HANDBOOK

FIREBERD 6000

REMOTE CONTROL COMMANDS

Telecommunications
Techniques
Corporation

...EXPECT EXCELLENCE

REGISTRATION

IMPORTANT

Please return the enclosed card so TTC may ensure that product updates, application notes, and new product information reaches the users of our equipment.

ML 10672
REV. C

FIREBERD 6000

REMOTE CONTROL COMMANDS HANDBOOK

FEBRUARY 1988

Telecommunications
Techniques
Corporation

© Telecommunications Techniques Corporation

444 NORTH FREDERICK AVENUE
GAITHERSBURG, MARYLAND USA
(301) 258-5011 TELEX: 908736

This Handbook applies to
FIREBERD 6000 Communications Analyzers
incorporating Software Revision D.

HANDBOOK ABSTRACT

This Remote Control Commands Handbook supplements the information found in the FIREBERD 6000 Operating Manual concerning remote control operation. The five sections of the Handbook are described below.

SECTION 1 PRINTER/CONTROLLER INTERFACES

This section provides an introduction to the FIREBERD 6000's printing and remote control capabilities. Specifically discussed are the built-in RS-232 Printer/Controller Interface and the IEEE-4888 Printer/Controller Interface and their operation for both printing and remote control purposes. This section also discusses the various print and graphics formats available.

SECTION 2 REMOTE CONTROL COMMANDS

This section contains all of the remote control commands used with the FIREBERD 6000 as well as command entry examples. Each command is listed on a separate page for easy reference.

SECTION 3 FIREBERD 6000 ERROR MESSAGES

This section contains a list and explanation of the error messages generated by the FIREBERD 6000 while under remote control operation.

SECTION 4 FIREBERD 6000 ON-LINE HELP DIRECTORY

This section presents the HELP pages that are available on-line for use with the FIREBERD 6000.

SECTION 5 SAMPLE INTERACTIVE CONFIGURATION SESSION

This section lists a sample interactive FIREBERD configuration session using the built-in BERD-BASIC programs.

SECTION 6 SAMPLE IEEE-488 PROGRAM

This section contains a sample IEEE-488 program.

TABLE OF CONTENTS

Section	Title	Page
1.0	**PRINTER/CONTROLLER INTERFACES**	1-1
1.1	Introduction	1-1
1.1.1	Printer Selection	1-2
1.1.2	Controller Selection	1-2
1.2	Printer and Graphics Formats	1-3
1.2.1	Results Print	1-3
1.2.2	Controls Print	1-3
1.2.3	Status Prints	1-7
1.2.4	Histogram Print	1-7
1.2.5	Jitter Frequency Prints	1-9
1.2.6	Print Formats	1-13
1.3	Remote Controller Commands	1-14
1.3.1	Terminal Operation and the On-line HELP Functions	1-14
1.3.2	Remote Control Command Formats	1-14
1.3.3	Remote Control Command Entry Sequence	1-15
1.3.4	FIREBERD Commands by Groups	1-15
1.3.5	Command Line Shortcuts	1-18
1.3.6	Pausing FIREBERD Printouts	1-18
1.3.7	Aborting FIREBERD Operations	1-19
1.4	BERD-BASIC Programming Concepts	1-19
1.5	RS-232 Printer/Controller Interface	1-21
1.5.1	Interface Description	1-21
1.5.2	Auto-baud Function	1-22
1.6	IEEE-488 Printer/Controller Interface	1-23
1.6.1	IEEE-488 Programming Hints	1-24
2.0	**REMOTE CONTROL COMMANDS**	2-1
3.0	**FIREBERD 6000 ERROR MESSAGES**	3-1
4.0	**FIREBERD 6000 ON-LINE HELP DIRECTORY**	4-1
5.0	**SAMPLE INTERACTIVE CONFIGURATION SESSION**	5-1
6.0	**SAMPLE IEEE-488 PROGRAM**	6-1

Digitized by the Internet Archive
in 2025

https://archive.org/details/manualsplus_10229

LIST OF FIGURES

Figure	Title	Page
1-1	Standard Results Print	1-4
1-2	Long Results Print	1-4
1-3	Custom Results Print	1-5
1-4	Delta Print	1-5
1-5	Controls Print	1-6
1-6	Status Prints	1-8
1-7	Histogram List Print	1-10
1-8	Histogram Graph Print	1-11
1-9	Jitter Frequency List	1-12
1-10	Jitter Frequency Graph	1-13
1-11	Sample BERD-BASIC Program	1-21

SECTION 1
PRINTER/CONTROLLER INTERFACES

1.1 INTRODUCTION

The FIREBERD 6000 is capable of communicating with external devices, such as printers and controllers, through its built-in RS-232 Printer/Controller Interface or the optional IEEE-488 Printer/Controller Interface. These printer/controller interfaces enable the FIREBERD to send all results and/or status information to an external printer or computer, allowing the user to obtain hard-copy printouts of all results and configurations. The printer/controller interfaces also allow a remote user to exercise full control over all normal FIREBERD functions (except power) and, additionally, they allow the user to gain access to many remote-control-only functions, including use of the BERD-BASIC programming language.

The FIREBERD printer/contoller interfaces are designed to communicate equally well with either a user operating a dumb terminal or a computer running an applications program. When communicating with a user operating a dumb terminal, the FIREBERD runs interactively; that is, it sends a prompt character to the terminal whenever it is ready to receive a command, and it echoes all command characters back to the terminal as the user types them. The FIREBERD also provides full on-line help to aid the user in entering commands. When communicating with a computer running an application program, the FIREBERD disables the prompt and echo functions. This allows the remote computer to concentrate on sending commands and getting results. Both of these modes are possible through either of the RS-232 or the IEEE-488 printer/controller interfaces.

The RS-232 Printer/Controller Interface is configured as data communications equipment (DCE), which allows it to be connected directly to a printer or to any other piece of data terminal equipment. All communications through the RS-232 port are asynchronous using the standard ASCII characters, with the exception of graphics codes used to produce output to graphics printers, such as the PR-85 printer. The baud rate and character format, as well as other RS-232 specific functions, can be configured using Auxiliary Function 38. The RS-232 Printer/Controller Interface is discussed in more detail in Section 1.5.

The optional IEEE-488 Printer/Controller Interface, which installs into a slot on the FIREBERD rear panel, complies with the IEEE Standard Digital Interface for Programmable Instrumentation (STD 488-1978). The interface offers both Talk-Only and Addressable modes. When operating in Talk-Only mode, the FIREBERD should be connected directly to a listen-only printer. In Addressable mode, the FIREBERD may be connected with up to 14 other devices, one of which must be a controller. Auxiliary Function 39 is used to select other IEEE-488 specific functions. The IEEE-488 Printer/Controller Interface is discussed in more detail in Section 1.6.

1.1.1 Printer Selection

The FIREBERD 6000 may be connected to an RS-232 external printer or may commun cate to a printer via the IEEE-488 interface. When both the RS-232 and IEEE-488 interfaces are installed, Auxiliary Function 11 is used to select which one the FIREBERD is to print to. The user may specify either that the FIREBERD always print to the RS-232, always print to the IEEE-488 or, alternately, that the FIREBERD decide which interface to print to. (The FIREBERD will select the RS-232 interface whenever it detects the DTR signal true; otherwise, it will select the IEEE-488.)

Auxiliary Function 11 is accessed through the AUXILIARY function of the MENU switch. When this function is selected, the following soft-key labels will be displayed.

(1) AUTO – Select RS-232 first; if no device connected (no DTR), then send to IEEE-488.

(2) RS-232 – Select RS-232 only.

(3) 488 – Select IEEE-488 only.

1.1.2 Controller Selection

The FIREBERD 6000 can operate in one of three control modes: Local, Remote, or Program. On power-up, the FIREBERD enters Local mode, in which the front panel switches control all functions. Upon reception of a valid remote control command from the controller, the FIREBERD enters Remote mode. In Remote mode, the front panel switches are ignored and all FIREBERD functions are controlled by the remote user. The remote user may elect to enter and run a BERD-BASIC program (or use a previously entered one), thus placing the FIREBERD in Program mode. While in Program mode, the FIREBERD receives all of its commands from the running program. This continues until the program stops, either by design or because of a break request from the remote controller. At this point, the FIREBERD returns to Remote mode and the remote user again has control over the instrument. The user may then return the FIREBERD back to Local mode, re-enabling front panel operation.

Only one interface may control the FIREBERD at a time Once either of the interfaces gains control (by sending a valid remote control comand), the other will be ignored until the FIREBERD is returned to Local mode. Once back in Local mode, both interfaces will be monitored, allowing either one the chance to take control of the FIREBERD.

1.2 PRINTER AND GRAPHICS FORMATS

The FIREBERD printouts can be divided into five broad categories: results prints, controls prints, status prints, histogram prints, and jitter frequency prints. Note that both the histogram and jitter frequency printouts may be either graphed on an Epson-compatible graphics printer or listed to a terminal. Each of these prints will be described in the following sections.

1.2.1 Results Print

Results prints may include some or all of the results accumulated by the FIREBERD. Results prints may be initiated by (1) a user request (pressing the RESULTS switch), (2) the occurrence of one of the user-selected print events, or (3) the occurrence of a results counter overflow. Note that, in a printout initiated because of a results counter overflow, the offending result will be marked with a double asterisk (**), indicating that it has just overflowed. In any subsequent printouts, this result will be marked with a single asterisk (*) indicating that it had previously overflowed.

The user may select which FIREBERD results are to be included in results printouts using Auxiliary Function 35. A long results print includes all available results. A standard results print, on the other hand, includes only an abbreviated list of the more commonly used results. If neither of these selections is suitable, a custom results print may be constructed in which the user may selectively include (or exclude) any of the available FIREBERD results.

If the Delta print function has been selected through the PRINT EVENT function of the MENU switch, automatic Delta printouts will occur at the end of each test interval. These printouts will contain only data accumulated over the last test interval. The Delta printout will not include average error rates, performance analysis results, or other long-term cumulative analysis results. In all other respects, the Delta printout will be treated in the same manner as any normal printout (with regard to format, results selections, etc.).

Figures 1-1 through 1-4 show sample results printouts.

1.2.2 Controls Print

The controls printout lists the current state of all FIREBERD mainframe controls. When applicable, the state of interface controls are listed also. This provides the user with enough information to duplicate the current configuration of the FIREBERD at another time or site.

Figure 1-5 shows a sample controls printout.

```
MANUAL          PRINT
05:07:32  04 MAR 87
SITE ID         Alpha
BIT ERRS            0
AVG BER      0. E-08
BER          0. E-06
BLOCKS        312012
BLK ERRS            0
E A SEC          4875
E F EAS          4875
RCV FREQ     64000.0
GEN FREQ     64000.0
%EFS              100%
IF  INTERNAL RS-232
```

Figure 1-1
Standard Results Print

```
MANUAL          PRINT
05:08:11  04 MAR 87
SITE ID         Alpha
BIT ERRS            0
AVG BER      0. E-08
BER          0. E-06
BLOCKS        314538
BLK ERRS            0
AVG BLER     0. E-05
E A SEC          4914
ERR EAS             0
E F EAS          4914
CLK SLIP            0
ELAP SEC         4915
DELAY
RCV FREQ     64000.0
GEN FREQ     64000.0
SYN LOSS            0
DAT LOSS            0
CLK LOSS            0
PWR LOSS            0
C-D CHA             0
SYNL SEC            0
%SYN SEC         100%
AVL SEC          4914
%DEG MIN         0.0%
%SES             0.0%
%EFS             100%
EFS              4914
DEG MIN             0
ERR SEC             0
SES                 0
UNA SEC             0
%AVL SEC         100%
IF  INTERNAL RS-232
DTR: OFF   RTS: OFF
DSR: OFF   CTS: OFF
RLSD: OFF
SELF LOOP ON
```

Figure 1-2
Long Results Print

```
MANUAL          PRINT
05:17:45  04 MAR 87
BIT ERRS            0
BER           0. E-04
BLK ERRS            0
CHAR ERR            0
ERR EAS             0
DELAY
SYN LOSS            0
DTR: OFF    RTS: OFF
DSR: OFF    CTS: OFF
RLSD: OFF
```

Figure 1-3
Custom Results Print

```
DELTA           PRINT
03:46:32  04 MAR 87
SITE ID        Alpha
BIT ERRS            0
BER           0. E-06
BLOCKS           1000
BLK ERRS            0
E A SEC            15
E F EAS            15
IF   INTERNAL RS-232
```

Figure 1-4
Delta Print

```
15:05:03       28 MAY 86
SITE:                Alpha
DATA:                MARK
ERROR INS:            OFF
SELF LOOP:             ON
GEN CLOCK:          SYNTH
TIMING MODE:        ASYNC
SYN FRQ:          9.6 kHz
INTERFACE:
 INTERNAL RS-232
 RI:                 OFF
 CTS:                OFF
EMULATE:             DTE
 DTR:                OFF
 RTS:                OFF
TEST INT:           10^4
ASYNC CHAR FORMAT:
 DATA BITS:             8
 STOP BITS              1
 PARITY:              ODD
JITTER:
 WDBAND:
  FILTER:           FULL
  THR:           6.5 UI
SPECTRUM ANALY:
 FREQ:             10 Hz
 SWEEP:              OFF
 MASK:             0.171
PRINT    EVENT:    DELTA
 TEST INT:           OFF
 SYNC LOSS:          OFF
 ERROR:              OFF
 TIME:               OFF
ANALY MODE:          CON
DISPLAY HOLD:        OFF
BLOCK LENGTH:
             100000 BITS
DELAY:    DTR/      DTR/
XON CHARACTER:         11
XOFF CHARACTER:        13
AUX FUNC IN USE:     NONE
REMOTE RS-232:
 DATA BITS:             8
 PRINT SPEED:        FAST
 BAUD:               4800
 PARITY:             EVEN
 TERM:              CR LF
```

Figure 1-5
Controls Print

1.2.3 Status Prints

The FIREBERD will automatically generate a status printout to inform the user of important developments with the ongoing test. Like the results in a results print, the various status messages may be individually enabled or disabled using Auxiliary Function 36. Most status printouts include a status message along with the current time/date and site identification name. These status messages include G.821 transition, data loss, clock loss, signal loss, sync loss, sync acquired, frame error (async character), receiver clock polarity change, test restart, new configuration, signaling change, power up, print buffer full, and results buffer full. Where applicable, these messages also include the count of the number of times they have occurred in the current test.

Figure 1-6 shows several sample status prints.

1.2.4 Histogram Print

A histogram print offers the user a perspective as to how a FIREBERD result is changing with time. For instance, a histogram could be used to monitor the accumulated bit errors in each hour over a 60-hour period. From this information, the user could determine whether the bit errors were uniformly distributed over the 60 hours or biased towards one point in time.

Histogram analysis of any integer results may be initiated and printed using Auxiliary Function 44. In this discussion, the applicable auxiliary function soft-key labels are shown in parentheses. The user must select the result to be monitored (RESULT), the units of the graph vertical axis (VERT), and the measurement interval (1, 10, or 60 minutes -- to be plotted along the horizontal axis) per histogram "bar" (HORIZ). Changing the measurement interval (HORIZ) or the result will cause the histogram analysis to restart; changing the units of the vertical axis will not cause a restart, thus allowing the data to be replotted with a new vertical scale.

The user may select either a graphics (GRAPH) or list (LIST) format for histogram prints. When printing graphics through the RS-232 interface, the user must be sure that the character format (selected with Auxiliary Function 38) is set for eight data bits. The histogram will be printed along with its start time, any sync loss indicators ("s") indicating intervals in which sync was lost, and any G.821 availability transition indicators ("t") indicating intervals in which transitions were made from available to unavailable time or from unavailable time to available time.

Once the histogram analysis is turned on, the FIREBERD will begin accumulating data for, at most, 60 intervals. When all intervals have been accumulated, the results will be printed and the histogram analysis will restart.

Figures 1-7 and 1-8 are examples of the histogram list and graph printouts.

```
NEW CONFIGURATION
SITE ID              Alpha
15:08:50      28 MAY 86

SYNC ACQUIRED           1
SITE ID              Alpha
15:08:81      28 MAY 86

CLOCK LOSS              1
SITE ID              Alpha
15:09:40      28 MAY 86

SYNC LOSS               1
SITE ID              Alpha
15:09:40      28 MAY 86

SYNC ACQUIRED           2
SITE ID              Alpha
15:09:41      28 MAY 86

LAST POWER DOWN
15:09:47      28 MAY 86
POWER UP
15:09:51      28 MAY 86
SITE ID              Alpha
```

Figure 1-6
Status Prints

1.2.5 Jitter Frequency Prints

When the DS1 Jitter Spectral Analysis option is installed, a "jitter frequency list" result is also available, either by itself or as part of a results print. This printout contains the jitter measurements for each of the 40 frequencies monitored by the spectrum analyzer. These measurements include the maximum unit intervals of jitter measured at each frequency along with the percentage of the selected jitter mask that this reading corresponds to. Again, like the Maximum Jitter result, when this result is included in a Delta print, it reflects only the measurements made during the test interval.

Jitter printouts may be initiated in either graphics or list format using the front panel JITTER function of the MENU switch. Note that when generating a graphics print through the RS-232 interface, the character format must be set for eight data bits.

Figures 1-9 and 1-10 are examples of the jitter frequency list and graph.

HISTOGRAM BEGAN AT
20:57:00 01 JAN 87

MINUTE	CHAR ERR
1	0
2	209
3	0
4	93
5	6
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	14
19	1
20	0
21	0
22	0
23	0
24	0
25	0
26	0
27	0
28	0
29	13
30	1
31	11
32	3
33	145

Figure 1-7
Histogram List Print

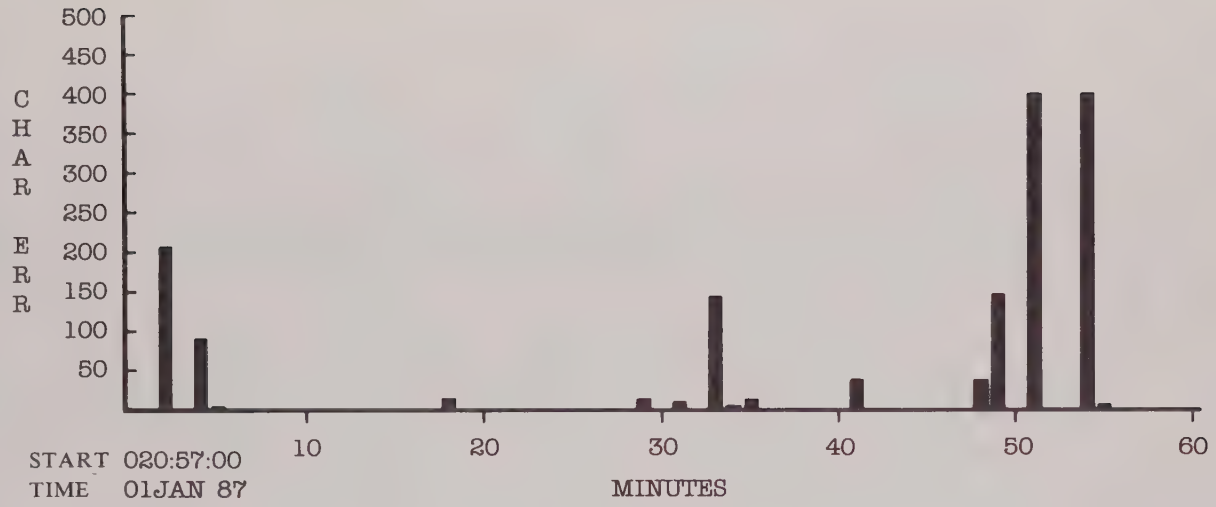


Figure 1-8
Histogram Graph Print

JITTER FREQ LIST :
15:10:23 01 JAN 87
MASK: O.171

FREQ(Hz)	UI	%MASK
10	0.04	0.4
20	0.03	0.3
30	0.02	0.2
40	0.05	0.5
50	0.02	0.2
60	0.01	0.1
70	0.01	0.1
80	0.01	0.1
100	0.02	0.2
133	0.02	0.2
166	0.03	0.3
200	0.02	0.2
233	0.03	0.3
300	0.13	1.9
400	1.06	20.9
500	8.15	201.2
600	0.44	13.0
700	0.14	4.8
800	0.10	3.9
1.0k	0.11	5.3
1.3k	0.05	3.2
1.6k	0.04	3.2
2.0k	0.03	2.9
2.3k	0.03	3.3
3.0k	0.03	4.3
4.0k	0.03	5.7
5.0k	0.05	11.9
6.0k	0.03	8.5
7.0k	0.03	10.0
8.0k	0.04	6.6
10k	0.04	13.3
13k	0.04	13.3
16k	0.01	3.3
20k	0.02	6.6
23k	0.01	3.3
26k	0.01	3.3
30k	0.01	3.3
33k	0.02	6.6
36k	0.02	6.6
40k	0.01	3.3

Figure 1-9
Jitter Frequency List

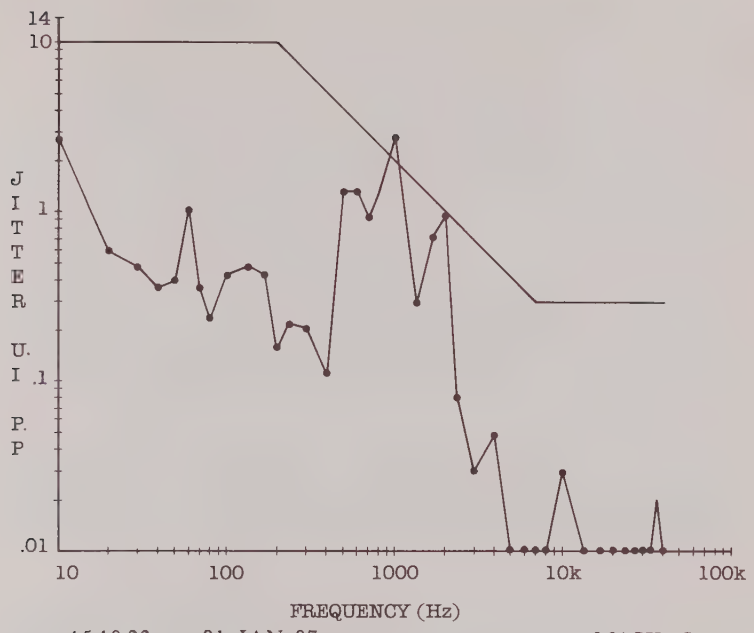


Figure 1-10
Jitter Frequency Graph

1.2.6 Print Formats

Auxiliary Function 37 may be used to specify the printout formats. This includes selecting the printout width (20 or 80 characters), printout spacing (condensed or normal), and printout order (forward -- first line first, or backwards -- last line first).

The soft-key labels for Auxiliary Function 37 are listed below.

(1) COLUMN – changes printout width from 20 to 80 columns.

(2) MODE – accesses soft-key labels COND (condensed) and NORMAL.

(3) LAYOUT – accesses soft-key labels FWD (forward) and BACKWD (backward).

The BACKWD setting should be selected for use with the PR-2000 20-column printer or other printers in which the paper is fed downward across the print mechanism. The FWD setting should be selected for use with CRTs, the PR-85 Printer, and most other printers.

1.3 REMOTE CONTROLLER COMMANDS

1.3.1 Terminal Operation and the On-Line HELP Functions

The FIREBERD 6000 provides extensive on-line help information intended to aid the user operating from a dumb terminal. This includes a 7 page, on-line, mini-manual as well as a parameter syntax guide for every remote control command.

To operate the FIREBERD from a dumb terminal, the user must first log in by typing a period (".") or the word "TERMINAL" followed by a carriage return. From there the FIREBERD will inform the user to type HELP followed by a carriage return to call up the main help screen. The main help screen will, in turn, inform the user of how to get more help on specific subjects. The pages of the on-line mini-manual are presented in Section 4 of this handbook.

From a dumb terminal, the user may then elect to enter interactive configuration mode. This is done using the built-in BERD-BASIC programs accessed via the **PROG** command (see section 2.0). Once in interactive mode, the FIREBERD will ask all questions necessary to configure a test, thereby alleviating the need for the user to memorize dozens of remote control commands.

Note that the log-in sequence is only to inform the FIREBERD that the controller is operating from a dumb terminal. This command should not be used when operating the FIREBERD from a computer, because it causes the FIREBERD to enable the prompt and echo functions, both of which will only confuse a computer program attempting to decipher the results returning from the FIREBERD.

1.3.2 Remote Control Command Formats

The general format for a remote control command is as follows:

command name (parameter...)

where **command name** is the command name as listed in Section 1.3.4 and **(parameter...)** is an optional parameter list associated with the command. Whenever possible, the remote control command and parameter names have been made consistent with those appearing on the FIREBERD's front panel. In general, remote control names are derived by shortening all words shown on the front panel to their first three letters, while leaving all numbers unchanged and representing exponents with a caret ("^").

Most FIREBERD commands may be used to either select a new command state or to print the currently selected command state (without changing it). To select a new command state, both the command name and the desired parameter are entered in the command line. To obtain the currently selected state, only the command name is entered. In other words, entering a command name without selecting a new state will cause the FIREBERD to print the currently selected state. This is referred to as a status request for the particular command.

1.3.3 Remote Control Command Entry Sequence

All commands may be entered to the FIREBERD in either upper or lower case, although the examples in this manual are listed in upper case only.

All space characters(" ") are ignored by the FIREBERD and thus may be used freely to make the commands more legible (as will also be done in all examples throughout this manual).

Entering a BACK-SPACE character (ASCII code 08H, Control H) causes the FIREBERD to discard the most recently entered character.

Entering a CANCEL character (ASCII code 18H, Control X) causes the entire command line currently being entered to be discarded.

All command lines to the FIREBERD must be terminated by a valid statement terminator. This may be a carriage return character (ASCII code 0DH), a linefeed character (ASCII code 0AH), or any character transmitted with the EOI line asserted (IEEE-488 only). Note that commands are not interpreted by the FIREBERD until this terminator is received.

1.3.4 FIREBERD Commands by Groups

Auxiliary Functions

AUX 1 – select/print generator clock polarity
AUX 2 – select/print receiver clock polarity
AUX 3 – select/print receiver action on sync loss
AUX 4 – select/print single transmit
AUX 5 – select/print data interface controller
AUX 6 – select/print generator data invert
AUX 7 – select/print receiver clock select
AUX 8 – select/print out-of-band flow control
AUX 9 – select/print in-band flow control
AUX 10 – select/print sync loss threshold
AUX 11 – select/print printer select
AUX 30 – select/print block length
AUX 31 – select/print delay measurement setup
AUX 32 – select/print common frequencies
AUX 33 – select/print programmable data pattern
AUX 34 – select/print site identification
AUX 35 – select/print custom results printout
AUX 36 – select/print custom status printouts
AUX 37 – select/print printout format
AUX 38 – select/print RS-232 printer/controller configuration
AUX 39 – select/print IEEE-488 printer/controller configuration
AUX 40 – print software revision level (see HELLO)

AUX 41 – set/print user message
AUX 42 – print pre-sync loss results
AUX 43 – clear all novram
AUX 44 – select/print histogram set-up
AUX 60 – set/print time and date
AUX 99 – clear auxiliary functions in use

Controller and General Configuration

BYE – return the FIREBERD to a power-up state (see DEVICE CLEAR)
DATE – set/print the date (see AUX 60)
DEVICE CLEAR – return the FIREBERD to a power-up state
ECHO – select/print controller echo mode
END – end an ENTER sequence
ENTER – disable contention checking on subsequent commands
HELLO – print the software revision level
HELP – print help information
ISU – control the Interface Switching Unit
LPRINT – print the literal string
LOCAL – return to FIREBERD local mode
MACRO – define user macro sequences
NUMBER – list command or parameter numbers
OPTIONS – list options currently installed in the FIREBERD
PAGE – print pages from the on-line mini-manual
PROG – use built-in BERD-BASIC programs
PROMPT – select/print the prompt mode
SELF TEST – print the results of the automatic self-test
TERMINAL – configure the FIREBERD to talk to a dumb terminal
TIME – set/print the time (see AUX 60)
TPZ – set state of rear panel test point "Z"

Display and Audio Indicator Functions

ANA RES 1 – select left display result
ANA RES 2 – select right display result
AUDIO – select/print audio level override
BEEP – sound the beeper
DIS HOL – select/print display hold mode
DISPLAY – select/print front panel display mode
SYN LOS – clear the SYNC/LOST LED

Interface Signaling, Etc.

DSR – manipulate/print DSR/(DM) control line state (DCE only)
DTR – manipulate/print DTR/(TR) control line state (DTE only)
LOO – generate loop codes (T-carrier only)
OCT TIM – control octet timing transmission on for 64 kb/s G.703 interface

RLS – manipulate/print RLSF/(RR) control line state (DTE only)
RTS – manipulate/print RTS/(RS) control line state (DTE only)
– use built-in BERD-BASIC programs

Printer Configuration

CUSTOM – select/print custom results prints (see AUX 35)
FORMAT – select formatted printer output (see AUX 37)
LONG – select long results prints (see AUX 35)
PLAIN ENGLISH – select character or hex user message prints
STANDARD – select standard results prints (see AUX 35)
STATUS – select/print custom status prints (see AUX 36)
TERM 232 – set/print RS-232 print line terminator (see AUX 38)
TERM 488 – set/print IEEE-488 print line terminator (see AUX 39)
UNFORMAT – select unformatted printer output (see AUX 37)
WIDTH – select/print printout width (see AUX 37)

Printer Functions

CLEAR FIFO – clear the printer buffer (see PRI)
CLS – clear the terminal screen
CONTROLS – generate a controls printout (see PRI)
HOLD – hold all printer outputs
LEDS – print front panel LED status
PRI – select/print printer enable/function
PRINT – print the single FIREBERD result or BERD-BASIC variable
REL – release all accumulated printer outputs from HOLD
RESULTS – generate a results printout (see PRI)

Test Configuration

ANA MOD – select/print analysis mode
CHA FOR – select/print async character format
DAT – select/print generator data pattern
ERR INS – select/print logic error insertion rate
ERROR – insert a single logic error (see ERR INS)
GEN CLK – select/print generator clock source
INT SET – select/print data interface configuration
JIT – select/print jitter measurement parameters
PRI EVE – select/print automatic result print events
REC – recall stored front panel program
RES – restart the test
SEL LOO – select/print self-loop mode
STO – store front panel program
SYN FRE – select/print internal synthesizer frequency
TES INT – select/print test interval
TIM MOD – select/print timing mode
USER – load user message with ASCII text

1.3.5 Command Line Shortcuts

A macro is a useful string of characters that can be inserted into a command at any point using just one or two keystrokes. The FIREBERD allows both user-defined and automatic macros. User macros are defined using the **MACRO** command and are recalled using the character sequence **& (number)** in the command line, where **(number)** is the macro identification number. The automatic help/status macro is set whenever a help (see the **HELP** command) or status request is issued for a particular command. The user can then refer to that command by simply using the "*" character in the command line. This saves the user the trouble of having to retype the command name again to alter its state.

Any command or command parameter issued to the FIREBERD may be abbreviated by entering its first few letters followed by a period ("."). The few letters of the command should be enough to uniquely identify it from all other FIREBERD commands or command parameters respectively. If a non-unique entry is generated in this manner, the FIREBERD will choose the first command or command parameter (as listed in the **HELP!** or **HELP(command name)** printouts respectively) that matches the entered abbreviation. For instance, **CU.** can be used as an abbreviation for the **CUSTOM** command since no other commands begin with the letters **CU.** On the other hand, **CO.** could not be used to abbreviate the **CONTROLS** command since the **COMMON** command also begins with the letters **CO** and the **COMMON** command appears first in the **HELP!** printout. (This means that **CO.** could be used to abbreviate the **COMMON** command; correspondingly, **CON.** could be used to abbreviate the **CONTROLS** command.)

Multiple parameters may be selected from the same command by separating them with commas (",") on the command line. For example, to set the asynchronous character format to 7 data bits and even parity, the controller could either enter the two commands **CHA FOR DAT 7** and **CHA FOR PAR EVE**, or enter the single command **CHA FOR DAT 7, PAR EVE.** Note that in the latter example, the **CHA FOR** command had only to be specified once and the comma separated the two parameter selections. One prompt will be returned to the controller for each parameter selected (preceded by an error message if the command was illegal).

Multiple commands may be issued to the FIREBERD on the same command line by separating them with semicolons (";"). The commands will be executed in the sequence in which they were entered. One prompt will be returned to the controller for each executed command (preceded by an error message if the command was illegal).

1.3.6 Pausing FIREBERD Printouts

Entering an X-OFF character (ASCII code 13H, Control S) will cause the FIREBERD to temporarily halt all printouts until either an X-ON character (ASCII code 11H, Control Q) or a statement terminator is received. Similarly, entering a space (" ") character will cause the FIREBERD to temporarily halt all

printouts, following the completion of the current line, until a statement terminator is received.

1.3.7 Aborting FIREBERD Operations

Entering a Control C character (ASCII code 03H, ETX) will cause the current input line to be discarded, the print buffer to be cleared, any **WAIT** or **LOO** commands in progress to be aborted, and any BERD-BASIC program currently executing to stop. This is the "break" command that gets the FIREBERD back to normal command mode.

1.4 BERD-BASIC PROGRAMMING CONCEPTS

A BERD-BASIC program is simply a sequence of commands that instruct the FIREBERD to perform desired functions. The program is written ahead of time and may be carried out later, possibly many times, without requiring the user to be present. The program itself takes control of the FIREBERD and, in effect, takes the place of the remote user.

A program needs to contain information as to both what commands are to be executed and in what order. This information is entered into the program as a sequence of **program lines**. Each program line is made up of two parts, a **line number** and a **command**. The **command** specified in a program line is the same remote control command that would be specified by the remote user; it tells the FIREBERD what to do. The **line number**, on the other hand, tells the FIREBERD when to do it. Program lines may be entered in any order; the FIREBERD will automatically sort them from lowest to highest line number as they are entered.

In general, a program will be executed starting with the lowest numbered line and proceeding, in sequence, to the highest numbered line. (For this reason, it is conventional to initially enter program lines numbered by tens, that is, 10, 20, 30, 40, etc. This ensures that several new lines can be easily inserted between two existing lines in case new commands need to be added later.) The absolute ordering of program line execution from lowest to highest line number may, however, be overridden. For example, the program can instruct the FIREBERD to "go back to" or "skip to" a specified line or, more importantly, the program can instruct the FIREBERD to "go to" a specified line only if a certain condition is true. In this way, the program can take certain actions depending on the values of one or more of the FIREBERD results.

BERD-BASIC also provides simple integer **variables** that can be used to count events. The program may print, change, or test the current value of any of these variables.

The FIREBERD allows the user to maintain up to five BERD-BASIC programs at a time — one "working" program and four stored on the ramdisk. The user can save and load programs to and from the ramdisk using easy-to-remember program names. Additionally, the running program may invoke the execution of any of the stored programs. Note that all five programs are stored in non-volatile RAM and, therefore, are not lost when the FIREBERD is powered down.

BERD-BASIC Program Commands

BREAK	– select program action on error
END	– end program execution
GOSUB	– transfer program execution to a subroutine
GOTO	– transfer program execution to specified line
IF-THEN	– conditional GOTO statement (based on result or variable value)
INPUT	– enter input from terminal into user macro
LET	– assign new value to BERD-BASIC variable
REM	– remark or comment
RETURN	– return from subroutine
STOP	– stop program execution (temporarily)
WAIT	– wait for sync or the specified time

BERD-BASIC Program Control Statements

CONT	– continue program execution from Control C or STOP
LIST	– list current BERD-BASIC program
RUN	– start program execution

BERD-BASIC Program Line Entry

(line number) (command)	– enter a program line
(line number)	– delete a program line

BERD-BASIC Ramdisk Operations

CHAIN	– chain execution to saved program
COMMON	– preserve all variable values through the next CHAIN
DIR	– list names of saved programs
KILL	– delete (unsave) a saved program from the ramdisk
LOAD	– load a saved program
SAVE	– save the current program on the ramdisk

Figure 1-11 shows an example of a BERD-BASIC program.

```
10 REM    = = = T1 NETWORK INTERFACE TEST = = = =
20 REC 3; WAIT 900; REM....REC = RECALL PROGRAM....900 SEC
   = 15 MINUTES
30 IF %MAS <= 100 THEN 50; REM....%MAS = MAX % OF
   JITTER SPECTRAL MASK
40 LPRINT "*** JITTER EXCEEDS AT&T PUB 62411 MASK ***"
50 IF BPVS <= 300 THEN 70; REM.....BPVS = BIPOLAR
   VIOLATIONS COUNT
60 LPRINT "*** MORE THAN 300 BPVS IN THE 15-MINUTE
   INTERVAL ***"
70 IF RCV FRE < 1543920 THEN 80; IF RCV FRE > 1544080
   THEN 80; GOTO 90
80 LPRINT "*** RECEIVED T1 FREQUENCY OUT OF
   TOLERANCE ***"
90 REM.....EQUIPMENT NEEDED — FIREBERD 6000 AND A
   DUMB TERMINAL.....
```

Figure 1-11
Sample BERD-BASIC Program

1.5 RS-232 PRINTER/CONTROLLER INTERFACE

1.5.1 Interface Description

The RS-232 Printer/Controller Interface is a built-in, standard feature of the FIREBERD 6000; the female, 25-pin D connector is located on the FIREBERD rear panel. The interface is configured such that the FIREBERD acts as DCE and may be directly connected directly to a printer (or any other piece of data terminal equipment). All communications through the RS-232 port are asynchronous using ASCII code, with the exception of Epson-compatible graphics codes used to produce output to graphics printers.

The following is a list of the RS-232 interface pins and their associated functions:

pin 1:	protective ground	– connected to chassis ground
pin 2:	transmit data	– data transmitted to the FIREBERD
pin 3:	receive data	– data received from the FIREBERD
pin 5:	clear to send	– output true when the FIREBERD is ready to receive
pin 6:	data set ready	– output true when the FIREBERD is on
pin 7:	signal ground	– connected to circuit ground
pin 8:	data carrier detect	– output true when the FIREBERD is on
pin 20:	data terminal ready	– must be input true before the FIREBERD will send

Note especially that the DTR signal (on pin 20) must be driven true by the printer before the FIREBERD will send any data.

Auxiliary Function 38 provides control over the RS-232 communication port configuration. Five categories of parameters may be controlled: baud rate; number of data bits per character; parity; line terminator; and print speed.

The baud rate may be set to 110, 300, 600, 1200, 2400, 4800, or 9600. The number of data bits may be either 7 or 8. The parity may be odd, even, or none. The line terminator transmitted from the FIREBERD may be set to carriage return (CR), linefeed (LF), both (CRLF), or none.

If the print speed is set to fast, the FIREBERD will send as fast as the baud rate allows. If the print speed is set to slow, the FIREBERD will send no more than 20 characters per second, and will pause for one-half second between lines.

The default settings for the FIREBERD are: 2400 baud, 7 data bits, even parity, carriage return, and slow print speed. If reconfigured, the FIREBERD will remember the new setings when powered down.

If the controller sends commands to the FIREBERD faster than it can process them (that is, if the controller ignores the Clear To Send output from the FIREBERD), the FIREBERD may not be able to keep up. If this occurs, the FIREBERD will respond with either a receiver overrun or receive buffer overflow error message and the command(s) will be discarded. To be safe, it is suggested that the controller either monitor the Clear To Send signal being output by the FIREBERD or allow the FIREBERD ample processing time following each command. (200 milliseconds is a good conservative figure for most commands.)

1.5.2 Auto-baud Function

To simplify the chore of setting up RS-232 remote control communications, the FIREBERD offers a very simple auto-baud function. This allows a remote user to configure the FIREBERD's baud-rate, data bits, and parity, without the trial-and-error approach normally associated with RS-232 communications. In fact, the remote user need not even know the configuration he/she is using; the FIREBERD will figure it all out automatically (and even tell the user, via AUX function 38).

To initiate auto-baud mode, the remote controller must send three **BREAK** characters (generally a separate key on the terminal that forces a steady space condition on the RS-232 line) at one second intervals. Following this, the user must repeatedly type **SPACE** characters (ASCII code 20H) at a five to ten hertz rate for ten seconds (or until the FIREBERD responds with the message "Auto-baud achieved, press <ESCAPE> to continue."). Once this is done, the user must send an **ESCAPE** (ASCII code 1BH) character and the FIREBERD will

respond with the message "Character format determined.". At this point, the FIREBERD is fully configured and the user may enter **TERMINAL** mode (if desired) and begin remote control.

There are a few items to note about the auto-baud procedure.

1) If auto-baud is not achieved within 30 seconds of being initiated, the FIREBERD will abort the function and revert back to its prior RS-232 settings. This timeout is necessary since many terminals and printers will generate a **BREAK** character as they are powered up or down, and thus might fool the FIREBERD into entering auto-baud mode when it was not supposed to. If this happens, the FIREBERD front panel display will flash the message "RS-232 auto-baud" until the 30 second timeout is finished, at which time the FIREBERD will abort auto-baud mode (bringing the RS-232 configuration back to what it was). Note also that changing AUX function 38 from the front panel will also abort the auto-baud function.

2) The **SPACE** characters sent to identify the terminals baud rate to the FIREBERD must be transmitted at such a rate that there is "dead-time" between characters on the RS-232 communications link. Usually, using the terminals "auto-repeat" function is fine; however, at 110 baud this may be too fast. If this is the case then the user may transmit the **SPACE** characters by hand at a minimum frequency of three per second. Note also that when auto-bauding via computer, the computer should send the **SPACE** characters at about 5 per second.

3) The message "Auto-baud achieved, press <ESCAPE> to continue." will always be transmitted with 7 data bits, 2 stop bits, and even parity, since the FIREBERD has not yet determined the character format of the remote controller. For this reason it may not appear normal on some terminals configured differently. The solution to this is to just go ahead and send the **ESCAPE** character anyway. The FIREBERD will then figure out the character format and everything will work properly.

4) While in auto-baud mode, the FIREBERD will constantly transmit a single asterisk ("*") and bell (Control G, ASCII code 07H) character at each baud rate that it scans. This is provided to give the user some indication that the FIREBERD is in auto-baud mode and that the user should be sending the **SPACE** characters.

1.6 IEEE-488 PRINTER/CONTROLLER INTERFACE

The FIREBERD IEEE-488 interface complies with the IEEE Standard Digital Interface for Programmable Instrumentation (STD 488 — 1978). The interface offers both Talk-Only and Addressable modes. When operating in Talk-Only mode, the FIREBERD should be connected directly to a listen-only

printer. In Addressable mode, the FIREBERD may be connected with up to 14 other devices, one of which must be a controller.

The rear panel DIP switch located on the IEEE-488 panel is used to select the interface mode along with the FIREBERD's interface address (used only in addressable mode). The DIP switch configuration is as follows:

	SW1	SW2	SW3	SW4	SW5	SW6
						Addressable
(0=DOWN, 1=UP)	A0	A1	A2	A3	A4	Talker Only

where Addressable/Talker Only selects the operating mode and A0 through A4 are the unit address. Although the FIREBERD IEEE-488 interface offers both Talk-Only and Addressable modes, remote control is available only in Addressable mode. Addtionally, when operating in addressable mode, the FIREBERD will respond to the Device Clear (DCL) or Group Execute Trigger (GET) commands by performing a re-power-up or test restart, respectively.

Auxiliary Function 39 provides control over the IEEE-488 communication port configuration. Three categories of parameters may be controlled: service request, line terminator, and print speed.

If service request is set to data, the FIREBERD will generate a service request (SRQ) whenever an erroneous command is received, or when it has data ready to transmit. If service request is set to error, the FIREBERD will generate a service request only when an erroneous command is received. In either case, the FIREBERD will set the most significant bit of its serial poll register (dav) whenever it has data to transmit. See Figure 1-12.

The line terminator transmitted from the FIREBERD may be set to carriage return (CR), linefeed (LF), both CRLF), or none. Regardless of mode or terminator selection, the EOI signal is raised with the final character of each line. (For software revision D and above only). If the print speed is set to fast, the FIREBERD will send as fast as the interface allows. If the print speed is set to slow, the FIREBERD will send no more than 20 characters per second, and will pause for one-half second between lines.

Figure 1-12
Serial Poll Status Byte

B7	B6	B5	B4	B3	B2	B1	B0
dav	rsv						syn

dav = data available in FIREBERD print buffer
rsv = request service (SRQ)
syn = command syntax error

The default settings for the FIREBERD 6000 are: error service request, CRLF line terminator, and fast print speed. If reconfigured, the FIREBERD will remember the new settings when powered down.

Unlike an RS-232 connection, the IEEE-488 bus requires that one device on the bus act as controller. All other devices connected to the bus must act as slaves to that controller. The FIREBERD can only act as a slave. Another intelligent device must be used as the bus as the bus controller.

Devices commonly used as a controller include the following:

- Hewlett-Packard HP-85 computer with the 82937 Interface.
- IBM-PC with a National Instruments GPIB-PC interface card and system software.

In most cases, a software program is required to operate the controller. Sample programs for the HP-85 and National Instruments GPIB-PC are shown in Section 6.

1.6.1 IEEE-488 Programming Hints

A device must be addressed to respond to a command. It is possible to address any single device or any group of devices on the bus.

Addressing a single device is accomplished with a statement such as Output (701); "DAT63" to the device at address 1 of the HP-85's port 7.

To address multiple devices using the HP-85, the REMOTE x,x,x statement is used. The x's constitute a list of devices to be addressed. Once the group of devices has been addressed, the group format of the commands may be used. One example is "Trigger 7," which causes all addressed devices on the bus to perform their DEVICE Trigger function.

Before attempting to read data from a device, it is necessary to know if the device has data to send.

The controller has two ways of determining that the FIREBERD has data: (1) bit 7 of the serial poll register (dav) is set whenever a line of data is available, and (2) if the FIREBERD AUX 39 SRQ mode selection is set to DAT, the FIREBERD makes a service request to the 488 controller whenever it has data available.

Both of these methods are demonstrated in the sample programs shown in Section 6.

The statement used to read data from the FIREBERD must be one that terminates the read operation when the last character of the line is encountered. Since it is up to the user to specify the FIREBERD's line terminator (CR, LF, CRLF), the most foolproof way to detect the last character is by sensing the EOI signal.

The HP-85 statement "ENTER 701 USING "#%,%K";A$" reads characters from device 1 until the EOI signal is encountered.

SECTION 2
REMOTE CONTROL COMMANDS

This section presents all of the remote control commands in alphabetical order. Each command is listed on a separate page; for ease of location, the command name appears in both the upper-left and upper-right corner of each page. Each page includes an explanation of the command, references to associated commands, and a brief example of command usage. Bold characters are used throughout this section to indicate the commands themselves as well as variables to be entered with the commands.

Enter a BERD-BASIC Program Line

(line number) (command)

Entering a BERD-BASIC program line number in front of a remote control command causes that command to be stored in the specified line of the working program. Any commands previously stored at the line are lost. Lines may be entered in any order; the FIREBERD will automatically sort them from lowest to highest line number. Note that commands entered into a program are not executed or checked for proper syntax until the program is **RUN**.

See Also: **RUN**

EXAMPLE:

```
> 10 BEEP                  : enter BEEP as line 10
> 30 PRINT BER             : enter PRINT BER as line 30
> 20 CLS                   : enter CLS as line 20
> LIST                     : then list the program
>

10 BEEP
20 CLS                     : (lines are kept in sorted order)
30 PRINT BER

[xxx bytes free]
>
```

Delete a BERD-BASIC Program Line

(line number)

Entering a BERD-BASIC line number without any associated commands following it deletes the corresponding line from the working program. Deleting a line that was not there to begin with has no effect.

EXAMPLE:

```
> 10 BEEP              : enter BEEP as line 10
> 30 PRINT BER         : enter PRINT BER as line 30
> 20 CLS               : enter CLS as line 20
> 10                   : delete line 10
> 30                   : delete line 30
> LIST                 : then list the program
>

20 CLS                 : (lines 10 and 30 have been deleted)

[xxx bytes free]
>
```

Analysis Mode

ANA MOD : print the analysis mode
ANA MOD [SIN|CON] : set the FIREBERD analysis mode

This command sets the FIREBERD to either **SIN**gle or **CON**tinuous analysis mode. In single analysis mode, the FIREBERD will only analyze a single **TES**t **INT**erval's worth of data for each test **RES**tart.

See Also: **TES INT, RES**

EXAMPLE:

```
> ANA MOD CON          : select continuous analysis mode
> ANA MOD              : then get the analysis mode status
>
CON

>
```

ANA RES 1 (result) : display **(result)** in left display window
ANA RES 2 (result) : display **(result)** in right display window

This command causes the FIREBERD to display the selected **(result)** in one of its result display windows. For a list of valid **(result)** names, please see the **PRINT** command. (Note that from this list, the **JTR FRE** and **(variable)** results may not be displayed.)

See Also: **PRINT**

EXAMPLE:

```
> ANA RES 1 BIT ERR      : put the bit error count on the left display
>
12
> ANA RES 2 TIM          : and the time of day on the right
>
12:48:00
>
```

Audio Level Override

AUDIO : print the audio level override status
AUDIO [ENABLE|DISABLE] : select the audio level override

This command allows the user to enable or disable the FIREBERD's audio indicator. When the **DISABLE** selection is chosen, no tones will be generated irrespective of the setting of the front panel volume control. When the **ENABLE** selection is chosen, the volume control will dictate how loud the tones will be.

EXAMPLE:

```
> AUDIO ENABLE        : enable the FIREBERD audio indicator
> BEEP                : (the FIREBERD will beep)
> AUDIO               : then print the current AUDIO status
>
ENABLE

> AUDIO DISABLE       : then disable the audio indicator
> BEEP                : (the FIREBERD no longer beeps)
>
```

Generator Clock Polarity

AUX 1 : print the generator clock polarity status
AUX 1 [NOR|INV] : set the generator clock polarity

This command sets the Generator Clock Polarity auxiliary function to **NOR**mal or **INV**erted. Note that **AUX 1 NOR** is the "cleared" state of this auxiliary function; **AUX 1 INV** is considered "in use".

See also: **AUX 5**

EXAMPLE:

```
> AUX 1 NOR          : set the generator clock polarity to normal
> AUX 1              : then display its status
>
NOR

>
```

Receiver Clock Polarity

AUX 2 : print the receiver clock polarity status
AUX 2 [AUT|NOR|INV] : select the receiver clock polarity

This command sets the Receiver Clock Polarity auxiliary function to either **AUT**o, **NOR**mal, or **INV**erted. Note that **AUX 2 AUT** is the "cleared" state of this auxiliary function; all other states are considered "in use".

See also: **AUX 5**

EXAMPLE:

```
> AUX 2 AUT          : select automatic receiver clock polarity
> AUX 2              : then get the receiver clock polarity status
>
AUT

>
```

Action on Sync Loss

AUX 3 : print the action on sync loss status
AUX 3 [CLE|HAL|CON|AUT DIS] : select action on sync loss

This command determines what action the FIREBERD will take upon detection of a loss of synchronization. The FIREBERD allows **CLE**ar, **HAL**t, **CON**tinuous, and **AUT**o sync **DIS**able modes. Note that **AUX 3 CLE** is the "cleared" state of this auxiliary function; all other states are considered "in use".

EXAMPLE:

```
> AUX 3 AUT DIS          : select auto sync disable mode
> AUX 3                  : then print the action on sync loss status
>
AUT DIS (*)

>
```

Single Transmit

AUX 4 : print the single transmit status
AUX 4 [OFF|ON] : select the single transmit mode

This command enables or disables the Single Transmit auxiliary function. If the **AUX 4 ON** selection is made, single transmissions are enabled and can be initiated with the **RES**tart command. Note that **AUX 4 OFF** is the "cleared" state of this auxiliary function; **AUX 4 ON** is considered "in use".

See Also: **RES**

EXAMPLE:

```
> AUX 4 ON          : go into single transmit mode
> AUX 4             : then print the single transmit status
>
ON (*)

> RES               : and initiate a single transmission
>
```

AUX 5 : print the data interface controller status
AUX 5 [F-BERD|INT] : select the data interface controller

This command selects whether the installed data interface will be controlled by the FIREBERD mainframe (**F-BERD**) or by the rear panel **INT**erface switches. This auxiliary function is only applicable to interfaces that can be controlled either by the mainframe or by their own switches. Note that **AUX 5 F-BERD** is the "cleared" state of this auxiliary function; **AUX 5 INT** is considered "in use".

```
> AUX 5 F-BERD          : set the interface under FIREBERD control
> AUX 1 NOR             : then control the interface's generator
> AUX 2 AUT             :   and receiver clock polarities
> AUX 5                 : then print the interface controller status
>
F-BERD
>
```

Generator Data Invert

AUX 6 : print the generator data invert status
AUX 6 [OFF|ON] : set the generator data invert mode

This command determines whether the data generated by the FIREBERD is of normal (inverter **OFF**) or inverted (inverter **ON**) polarity. Note the **AUX 6 OFF** is the "cleared" state of this aux function; **AUX 6 ON** is considered "in use".

EXAMPLE:

```
> AUX 6                    : find the current generator data invert status
>
ON (*)

> AUX 6 OFF                : then change it to normal polarity
>
```

Receiver Clock Select

AUX 7 : print the receiver clock select status
AUX 7 [AUT|ST|TT] : select the source of the receiver clock

This command sets the FIREBERD receiver clock selection to **AUT**o, **S**end **T**iming, or **T**erminal **T**iming. Note that **AUX 7 AUT** is the "cleared" state of this auxiliary function; all other states are considered "in use".

EXAMPLE:

```
> AUX 7 AUT          : set automatic receiver clock selection
> AUX 7              : query the receiver clock selection status
>
AUT
>
```

AUX 8	: print the out-of-band flow status
AUX 8 OFF	: disable all out-of-band flow control
AUX 8 ON [TR|DM|RS|CS|RR]	: enable the out-of-band flow control

This command allows the user to specify which signaling lines should be used for out-of-band flow control. RS-449 signal names must be specified. The relationship between RS-449 and RS-232 signal names is as follows:

RS-449	RS-232
TR	DTR
DM	DSR
RS	RTS
CS	CTS
RR	RLSD

More than one signal line may be selected simultaneously. Note that **AUX 8 OFF** is the "cleared" state of this auxiliary function; all other states are considered "in use".

EXAMPLE:

```
> AUX 8 OFF          : disable all out-of-band flow control
> AUX 8              : and print the out-of-band flow control status
>
OFF

> AUX 8 ON RS        : enable flow control on the RS signal
> AUX 8 ON TR        :   and on the TR signal
> AUX 8              : print the out-of-band flow control status
>
ON (*) RS            : (both RS and TR are ON)
       TR
>
```

In-Band Flow Control

AUX 9	: print the in-band flow status
AUX 9 OFF	: disable all in-band flow control
AUX 9 ON [XON(hex)|XOFF(hex)]	: set the in-band flow control

This command selects the in-band flow control mode. If **AUX 9 OFF** is selected, no in-band flow control takes place. If **AUX 9 ON** is selected, in-band flow control will take place with the selected **XOFF** (halt data transmission) and **XON** (restart data transmission) characters. These characters are selected with the **AUX 9 ON XOFF (hex)** and **AUX 9 ON XON (hex)** commands respectively, where **(hex)** is the hexadecimal code (from 00 to FF) of the character to be used. Note that **AUX 9 OFF** is the "cleared" state of this auxiliary function; all other states are considered "in use".

EXAMPLE:

```
> AUX 9 OFF                    : disable in-band flow control
> AUX 9                        : then print the in-band flow control status
>
OFF

> AUX 9 ON                     : then enable in-band flow control
>
XON 11 (hex)                   : (currently, in ASCII, XON=DC1
                                   and XOFF=DC3)
XOFF 13 (hex)

> AUX 9 ON XON 0D              : and change the XON character to an ASCII(CR)
> AUX 9                        : and print the in-band flow status
>
ON (*) XON 0D (hex)
       XOFF 13 (hex)
>
```

Sync Loss Threshold

AUX 10 : print the sync loss threshold status
AUX 10 [MED|LOW|HIG] : set the sync loss threshold

This command sets the error rate threshold required for the FIREBERD to declare a loss of synchronization. It can be set to **LOW** (100 errors in 1000 bits), **MED**ium (250 errors in 1000 bits), or **HIG**h (20,000 errors in 100,000 bits). Note that **AUX 10 MED** is the "cleared" state of this auxiliary function; all other states are considered "in use".

EXAMPLE:

```
> AUX 10 MED          : set medium sync loss threshold
> AUX 10              : print the sync loss threshold status
>
MED
>
```

Printer Select

AUX 11 : print the printer select status
AUX 11 [AUT|RS-232|488] : set the printer select function

This command determines where the printer results will be sent by the FIREBERD. If the **AUT**o selection is chosen, printouts will go to the RS-232 port if the data terminal ready signal is received true. Otherwise, printouts will go to the IEEE-488 (if installed). Selecting either RS-232 or 488 will force the printouts to go to the corresponding port regardless of the state of the DTR signal line. Note that **AUX 11 AUT** is the "cleared" state of this auxiliary function; all other states are considered "in use".

EXAMPLE:

```
> AUX 11                    : print the current printer select status
>
AUT

> AUX 11 RS-232             : force printouts to the RS-232
> AUX 11                    : then get the new status
RS-232 (*)
>
```

Block Length

AUX 30 : print the block length status
AUX 30 (length) : set new block length

This command sets the number of bits in each block (as referred to by the block analysis results). The **(length)** can be any integeral number of bits from 100 to 1000000.

EXAMPLE:

> AUX 30 : print the current block length
>
1000 bits

> AUX 30 12345 : set a new block length
>

AUX 31 : print the delay measurement status
AUX 31 START (event) : set the delay measurement start event
AUX 31 STOP (event) : set the delay measurement stop event

This command defines the **delay** interval to measure. The delay measurement will start upon the detection of the **START (event)** and will stop upon the detection of the next **STOP (event)**. The result of the delay measurement is available through the **RESULTS** printout or though the **PRINT DEL** command. The valid **START** and **STOP (event)**s are as follows:

DTR/ : rising edge of interface DTR signal
DTR : falling edge of interface DTR signal
DSR/ : rising edge of interface DSR signal
DSR : falling edge of interface DSR signal
RTS/ : rising edge of interface RTS signal
RTS : falling edge of interface RTS signal
CTS/ : rising edge of interface CTS signal
CTS : falling edge of interface CTS signal
RLS/ : rising edge of interface RLSD signal
RLS : falling edge of interface RLSD signal
TP/ : rising edge of rear panel test point
TP : falling edge of rear panel test point
GPA or **RPA** : generator or receiver pattern (**START** or **STOP** respectively)
GDA/ or **RDA/** : generator or receiver data (**START** or **STOP**) rising edge
GDA or **RDA** : generator or receiver data (**START** or **STOP**) falling edge

See also: **PRINT, RESULTS**

EXAMPLE:

```
> AUX 31                    : print the current delay status
>
START RTS/                  : (measure from rising to falling edge of RTS)
STOP RTS\

> AUX 31 STOP CTS/          : change to measure from RTS/ to CTS/
> AUX 31                    : and print out the new delay status
>
START RTS/
STOP CTS/
>
```

Common Frequencies

AUX 32	: print all of the common frequencies
AUX 32 (number)	: print the common frequency #(number)
AUX 32 (number)/ (frequency)	: set common frequency #(number) to(frequency)

This command allows the user to change the "common frequencies" that show up on the front panel when selecting the synthesizer frequency. The **(frequency)** field is any valid frequency (in kHz) in the range .05kHz to 15000 kHz. The frequency **(number)**s range from 1 to 11, with 1 being the left-most frequency displayed when the MENU switch is first moved to the SYNTH FREQ position.

EXAMPLE:

```
> AUX 32                    : print the current common frequencies
>
   1/1.2 (kHz)
   2/2.4 (kHz)
   3/4.8 (kHz)
   4/9.6 (kHz)
   5/19.2 (kHz)
   6/56 (kHz)
   7/64 (kHz)
   8/112 (kHz)
   9/1544 (kHz)
  10/3512 (kHz)
  11/6312 (kHz)

> AUX 32 1/.110             : then set common frequency 1 to 110 bits/sec
> AUX 32 1                  : display the setting of common frequency 1
>
.110 (kHz)
>
```

Programmable Data Pattern

AUX 33 : print the programmable data pattern status
AUX 33 (pattern) : set the programmable data pattern

This command allows the user to define the data pattern that is generated when the **DAT**a switch is in the **PRG**m position. The user-defined **(pattern)** is any sequence of 3 to 24 ones (1's) and/or zeros (0's), corresponding to the desired binary bit pattern, with the left-most bit transmitted first.

See also: **DAT**

EXAMPLE:

```
> AUX 33 11010011      : set the programmable pattern to "11010011"
> AUX 33               : print out the programmable pattern
>
11010011
>
```

AUX 34 : print the current site ID
AUX 34 (string) : set the site ID

This command updates the site ID that is printed along with most status and results prints. The site ID **(string)** may be from 1 to 10 characters in length.

EXAMPLE:

```
> AUX 34                    : print the current site ID
>
Alpha

> AUX 34 BETA               : change the current site ID
> AUX 34                    : and print out what it is now
>
BETA
>
```

Custom Results Printout

AUX 35	: print the custom results print status
AUX 35 STD	: select standard results print
AUX 35 LON	: select long results print
AUX 35 CUS	: select customized results print
AUX 35 CUS (result) [ON | OFF]	: enable or disable a custom result
AUX 35 CUS ALL [ON | OFF]	: enable or disable all custom results
LONG	: alternate form of **AUX 35 LON**
STANDARD	: alternate form of **AUX 35 STD**
CUSTOM ...	: alternate form of **AUX 35 CUS ...**

This command is used to tailor the results printouts generated by the FIREBERD. If the **ST**an**D**ard option is chosen, the FIREBERD will print an abbreviated list of the more important results. If the **LON**g option is chosen, all available results will be printed. If neither of these options is suitable, the **AUX 35 CUS** option may be used in which the printout of each individual **(result)** may be selectively enabled or disabled by choosing **AUX 35 CUS (result) ON** or **AUX 35 CUS (result) OFF** respectively. The **(result)** names to be customized may be any of the following:

%ACT SEC	: percentage of active seconds
%AVL SEC	: percentage of available seconds
%DEG MIN	: percentage of degraded minutes
%EFS	: percentge of error-free seconds
%MAS	: maximum percentage of jitter mask
%SES	: percentage of severely errored seconds
%SYN SEC	: percentage of sync seconds
%SVS	: percentage of severely violated seconds
%VF EAS	: percentage of violation-free error analysis seconds
%VFS	: percentage of violation-free seconds
1SEC CRC	: one second CRC errors
1 SEC JTR	: maximum unit intervals of jitter over previous second
AIS SEC	: AIS seconds
AVG BER	: average bit error rate
AVG BPVR	: average bipolar violation rate
AVG BLER	: average block error rate
AVG CRC	: average CRC error rate
AVG FER	: average framing error rate
AVG FWER	: average frame word error rate
AVL SEC	: available seconds
BER	: bit error rate
BIT ERR	: bit errors
BLK ERR	: block errors
BLO	: blocks
BPV RAT	: bipolar violation rate

BPVS	: bipolar violations
C-D CHA	: clock-data phase changes
CHA ERR	: asynchronous character errors
CLK LOS	: clock losses
CLK SLI	: clock slips
CLO INV	: clock invert message
CRC E RT	: CRC error rate
CRC ERR	: CRC errors
DAT INV	: data invert message
DAT LOS	: data losses
DEG MIN	: degraded minutes
DEL	: delay measurement
EA SEC	: error analysis seconds
EF EAS	: error-free error analysis seconds
EFS	: error-free seconds
ELA SEC	: elapsed test seconds
ERR EAS	: errored error analysis seconds
ERR SEC	: errored seconds
FE RAT	: framing error rate
FRA ERR	: framing errors
FRA LOS	: frame sync losses
FW ERR	: frame word errors
FW E RATE	: frame word error rate
GEN FRE	: generator clock frequency
IF	: interface type
JTR FRE	: jitter frequency list
JTR HIT	: jitter hits
MAX JTR	: maximum unit intervals of jitter
PWR LOS	: power losses
RCV COD	: received line code
RCV FRE	: receiver clock frequency
SEL LOO	: self loop message
SES	: severely errored seconds
SIG LOS	: signal losses
SIGL SEC	: signal loss seconds
SIG	: interface signaling
SIT ID	: FIREBERD site identification
SVS	: severely violated seconds
SYN LOS	: sync losses
SYNL SEC	: sync loss seconds
TIM/DAT	: time and date line
UNA SEC	: unavailable seconds
VF EAS	: violation-free error analysis seconds
VFS	: violation-free seconds
XSO SEC	: excess zero seconds
YEL SEC	: yellow alarm seconds

Note that all custom results may be enabled or disabled at once using the **AUX 35 CUS ALL ON** or **AUX 35 CUS ALL OFF** commands, respectively.

See also: **RESULTS, PRI**

EXAMPLE:

```
> LONG                    : select long results print
> RESULTS                 : get a results printout
>
MANUAL     PRINT
  .
  .                        : (all results included)
  .
> AUX 35 CUS ALL OFF      : select custom print and disable all results
> CUSTOM BIT ERR ON       : then enable just the BIT ERR result
> RESULTS                 : and get a results print
>
MANUAL     PRINT BIT ERR     71
>
```

AUX 36 : print the custom status printout status
AUX 36 ALL [ON | OFF] : enable or disable all status printouts
AUX 36 (status) [ON | OFF] : enable or disable the (status) printout

STATUS ... : alternate form for **AUX 36 ...**

This command determines which of the FIREBERD status printouts will be enabled. If the **AUX 36 ALL ON** selection is made, all of the status printouts will be enabled. If **AUX 36 ALL OFF** is chosen, no status printouts will be generated. The **AUX 36 (status)** command allows each status printout to be selectively enabled or disabled by choosing either the **AUX 36 (status) ON** or **AUX 36 (status) OFF** commands, respectively. The **(status)** printout names are as follows:

CLK LOS : clock loss
DAT LOS : data loss
FRA ERR : framing error
G.8 TRA : G.821 transition
NEW CON : new configuration
OVE PRI : overflow printout
POL CHN : receiver clock polarity change
POW UP : power up
SIG CHA : signaling change
SIG LOS : signal loss
SYN ACQ : sync acquired
SYN LOS : sync loss
TES RES : test restart

Note that status prints will only be generated if the main **PRI**nter enable is set **ON**.

See also: **PRI**

EXAMPLE:

```
> PRI ON                    : main printer enable
> STATUS ALL OFF            : disable all status messages
> RES                       : restart the test (no status print generated)
> AUX 36 TES RES ON         : then enable the test restart status messages
> RES                       : and now restart the test
>
TEST RESTART
  .                         : (now status printout is generated)
  .
  .
>
```

AUX 37	: print the print format status
AUX 37 COL [20 | 40 | 80]	: select the printer paper/screen width
AUX 37 MOD [NOR | CON]	: select the print formatting
AUX 37 LAY [FWD | BWD]	: select the printout direction

WIDTH ...	: alternate format for **AUX 37 COL...**
FORMAT	: alternate format for **AUX 37 MOD NOR**
UNFORMAT	: alternate format for **AUX 37 MOD CON**

This command allows the user to configure the printout format. The printer width selection (**20**, **40**, or **80** characters per line) determines how the results and controls prints (along with some of the help printouts) will be printed across the page. The **NOR**mal or **CON**densed selections specify either formatted or unformatted printing, respectively. Unformatted printouts have an assumed printer width of 20 characters (limiting results and controls prints to one result per line), contain no spaces, and substitute result numbers for result names (as listed by the **NUMBER PRINT** command). The **F**or**W**ar**D** and **B**ack**W**ar**D** selections determine if results and controls prints should come first line first or last line first, respectively.

See also: **NUMBER**

EXAMPLE:

```
> WIDTH 20                : select a 20-column printer width
> DIR                     : get a directory of stored BERD-BASIC programs
>
"MYPROG1"
"FOO"                     : (notice all less than 20 characters per line)
"DAISY"

> WIDTH 40                : then select a 40 column printer width
> DIR                     : and get the same printout
>
"MYPROG1"    "FOO"
"DAISY"                   : (now we have up to 40 characters per line)

> AUX 37 MOD CON          : then go to condensed printouts
> CUSTOM ALL OFF          : and make a custom results print
> CUSTOM BIT ERR ON       : with just bit errors
> CUSTOM ELA SEC ON       : and elapsed seconds
> RESULTS                 : then get a results print
```

AUX 37

```
>
MANUALPRINT          : (notice no spaces)
17)19                : (19 bit errors)
33)73                : (73 elapsed seconds)
SELFLOOP ON          : (FIREBERD in self-loop)

> AUX 37 MOD NOR     : back to normal printouts
>
```

AUX 38	: print the configuration status
AUX 38 BAU (baud)	: set the RS-232 baud rate
AUX 38 DAT [7 | 8]	: set the number of data bits
AUX 38 PAR [EVE | ODD | NON]	: set the parity bit
AUX 38 TER [CR|LF| CRLF| NON]	: set the line terminator
AUX 38 PRI [FAS | SLO]	: set the print speed

TERM 232 ...	: alternate format for **AUX 38 TER ...**
FAST	: alternate format for **AUX 38 PRI FAS**
SLOW	: alternate format for **AUX 38 PRI SLO**

These commands configure the RS-232 port to communicate properly with a printer or controller. The (baud) rate specified may be any one of the following:

110, 300, 600, 1200, 2400, 4800, or **9600**

The port can be configured for **7** or **8** data bits and the parity selection may be either **EVE**n, **ODD**, or **NON**e. The line terminator is the sequence transmitted at the end of each printed line. It may be selected as **C**arriage **R**eturn, **L**ine**F**eed, **C**arriage **R**eturn and **L**ine**F**eed, or **NON**e. The print speed may be set to either **FAS**t (as fast as the FIREBERD can print) or **SLO**w (no more than 20 characters per second with an additional one-half second delay at the end of each printed line). Note that changing character formats or baud rate from a terminal may result in losing communications with the FIREBERD. From a BERD-BASIC program, however, the baud rates, etc., may be temporarily changed (say, long enough to get a results print from a slow printer) and then returned to the previous configuration to talk to the terminal. The example will show this.

EXAMPLE:

```
> AUX 38 BAU          : check the current baud rate
>
9600                  : (terminal operating at 9600 baud)

> 10 LPRINT "SWITCHING TO 300 BAUD"
> 15 LPRINT "PLEASE CONNECT PRINTER"
> 20 AUX 38 BAU 300   : switch over to the printer's speed
> 30 WAIT 30          : wait 30 seconds for user to connect printer
> 40 RESULTS          : then generate a results print to printer
> 50 WAIT 60          : wait for the results print to finish
> 55 BEEP             : then beep to get the users attention
> 57 LPRINT "PLEASE RECONNECT TERMINAL"
```

```
> 60 WAIT 30                    : then wait for the user to reconnect terminal
> 70 AUX 38 BAU 9600            : then switch back to the terminal's speed
> 80 LPRINT "BACK AT 9600 BAUD"
> 90 END                        : and let the user give commands again
> RUN                           : then the user runs the program
SWITCHING TO 300
  BAUD                          : (the printout goes to the printer and
PLEASE CONNECT
  PRINTER                       : the user eventually reconnects terminal)
BACK AT 9600 BAUD
>
```

AUX 39	: print the IEEE-488 configuration status
AUX 39 SRQ [DAT|ERR]	: select reason to generate SRQ
AUX 39 TER [CR|LF|CRLF|NON]	: select print line terminator
AUX 39 PRI [FAS|SLO]	: select printer speed
TERM 488 ...	: alternate format for **AUX 39 TER ...**

These commands allow the user to configure the IEEE-488 controller/printer port as desired. If the **AUX 39 SRQ ERR** selection is made, the FIREBERD will only generate a service request (SRQ) to indicate that the previously received command was in error. Otherwise, with **AUX 39 SRQ DAT**, the FIREBERD will generate a service request (SRQ) whenever it has data to transmit or whenever the previous command was in error. The **AUX 39 TER** command selects the line terminator to be appended to the end of every printed line. It may be selected as **C**arriage **R**eturn, **L**ine**F**eed, **C**arriage **R**eturn and **L**ine**F**eed, or **NON**e. The print speed may be set to either **FAS**t (as fast as the FIREBERD can print) or **SLO**w (no more than 20 characters per second with an additional one-half second at the end of each printed line).

EXAMPLE:

```
> AUX 39                    : print curent IEEE-488 configuration status
>
SRQ ERR
TER LF
PRI FAS

> AUX 39 TER CRLF           : set terminator to carriage return, linefeed
>
```

User Message Select

AUX 41	: print selected user message and contents
AUX 41 [1|2|3]	: select new user message
AUX 41 [1|2|3]/(user)	: load a user message with data

This command allows the user to select and modify user message **1**, **2**, or **3**. When loading a user message, **(user)** is a hex list of the user message bytes.

See also: **USER**

EXAMPLE:

> AUX 41	: see what user message is selected
>	
1/48 45 4C 4C 4F	: (user message 1, which is "HELLO" in ASCII)
> AUX 41 2/48 49	: set user message 2 to "HI" in ASCII and select it
> AUX 41	: then see what user message is selected
>	
2/48 49	: (it's what we just set)
>	

AUX 42

This command causes the pre-sync loss results to be printed and loaded into the display buffers. It also activates a display hold so that the pre-sync loss results may be examined for an extended period of time.

See also: **DIS HOL**

EXAMPLE:

```
> AUX 42                    : recall the pre-sync loss results
>
PRE SYNC LOSS RESULTS
  .
  .                         : (a pre-sync loss results print is generated)
  .

> DIS HOL OFF               : release the display hold
>
```

Clear All NOVRAM

AUX 43 : attempt to clear NOVRAM

This command clears all non-volatile RAM (NOVRAM) in the FIREBERD. It should be used with extreme caution since it wipes out all configurations, auxiliary functions, front panel programs, BERD-BASIC programs, etc. Since this is such a potentially dangerous command, successfully clearing NOVRAM requires entering a second command to verify your intentions. This is shown in the example.

EXAMPLE:

```
> AUX 43                    : attempt to clear novram
WARNING:  To really clear novram, type "AUX 43!"

> AUX 43!                   : then do it for real
>
```

AUX 44	: print the histogram setup status
AUX 44 PRI	: initiate a histogram print
AUX 44 [ON|OFF]	: enable or disable histograms
AUX 44 [GRA|LIS]	: select list or graph output
AUX 44 HOR [1 MIN|10 MIN| 60 MIN]	: select horizontal sample time
AUX 44 VER (units)	: select units for vertical axis
AUX 44 RES (result)	: select histogram result

This command is used to configure the histogram setup. Histogram analysis is enable using the **AUX 44 ON** command. Likewise, **AUX 44 OFF** disables all histogram analysis. Before any histogram analysis can be performed, the histogram **HOR**izontal and **VER**tical axes must be selected as well as the **RES**ult to be monitored. The **HOR**izontal (i.e., the time between samples) can be specified as either **1 MIN, 10 MIN,** or **60 MIN.** The **(units)** of the **VER**tical axis can be selected as one of the following:

1, 2, 5, 10, 20, 50, 100, 200, 500, 1K, 2K, 5K, 10K, 20K, 50K, 100K, 200K, 500K, 1M, 2M, or **5M**

where "K" and "M" stand for "thousand" and "million" respectively. The **(result)** to be monitored may be one of the following:

AIS SEC	: AIS seconds
AVL SEC	: available seconds
BIT ERR	: bit errors
BLK ERR	: block errors
BLO	: blocks
BPVS	: bipolar violations
C-D CHA	: clock-data phase changes
CHA ERR	: asynchronous character errors
CLK LOS	: clock losses
CLK SLI	: clock slips
CRC ERR	: CRC errors
DAT LOS	: data losses
DEG MIN	: degraded minutes
EAS	: error analysis seconds
EF EAS	: error-free error analysis seconds
EFS	: error-free seconds
ERR EAS	: errored error analysis seconds
ERR SEC	: errored seconds
FRA ERR	: framing errors
FRA LOS	: frame sync losses
FW ERR	: frame word errors
JTR HIT	: jitter hits

SES	: severely errored seconds
SIG LOS	: signal losses
SIGL SEC	: signal loss seconds
SVS	: severely violated seconds
SYN LOS	: sync losses
SYNL SEC	: sync loss seconds
UNA SEC	: unavailable seconds
VF EAS	: violation-fee error analysis seconds
VFS	: violation-free seconds
XSO SEC	: excess zero seconds
YEL SEC	: yellow alarm seconds

To generate graphics output to a graphics printer, the **AUX 44 GRA** function must be selected. Otherwise, using **AUX 44 LIS**, only a histogram listing (suitable for display on a terminal) will be generated. The **AUX 44 PRI** command is used to initiate the printout of the accumulated histogram data.

EXAMPLE:

```
> AUX 44 LIS              : select a histogram list
> AUX 44 HOR 1 MIN        : take samples every minute
> AUX 44 VER 1            : measure single bit errors on the
> AUX 44 RES BIT ERR      : vertical axis
> AUX 44 ON               : then turn on the histogram analysis
>                         : then wait 10 minutes
> AUX 44 PRI              : and get a histogram list
>
HISTOGRAM LIST
   .
   .                      : (list of bit errors versus minutes)
   .
>
```

Time and Date

AUX 60	: print the time and date
AUX 60 TIM (hh:mm)	: set the time of day
AUX 60 DAT (dd/mm/yy)	: set the date
TIME ...	: alternate form of **AUX 60 TIM ...**
DATE ...	: alternate form of **AUX 60 DAT ...**

This command sets and displays the **TIM**e and **DAT**e. The time is entered in 24-hour format and the seconds are always set to zero.

EXAMPLE:

```
> TIME                    : print the time
>
19:40:11                  : (7:40 PM)

> AUX 60                  : print the time and date
>
TIM 19:40:20
DAT 12 APR 88

> AUX 60 DAT 14/4/88      : then change the date to the 14th of April
> DATE 13/4/88            : and back to the 13th of April
>
```

AUX 99 : attempt to clear auxiliary functions in use

This command sets auxiliary functions 1 through 11 to their "cleared" state. Since this is a potentially destructive command, the FIREBERD attempts to verify the user's intention by requiring an additional command to actually clear the auxiliary functions. This is shown in the example.

EXAMPLE:

```
> AUX 99                    : attempt to clear auxiliary functions in use
WARNING: To really clear aux functions, type "AUX 99!"

> AUX 99!                   : now we really clear all auxiliary functions
>
```

Sound the FIREBERD's Beeper

BEEP

This command causes both the terminal's bell (Control G, ASCII code 07H) to ring and the FIREBERD's audio indicator to sound a single short beep. Note that no beep will generated if either the front panel volume control is set to off or the **AUDIO DISABLE** remote control function has been selected.

See also: **AUDIO**

EXAMPLE:

```
> BEEP                    : sound the beeper once
> BEEP                    : and again
>
```

BREAK [ON|OFF|RESET|WARN]

This command determines how the FIREBERD will respond to errors made in a running program. The default selection, **BREAK ON**, will cause the FIREBERD to break program execution immediately after an error occurs. If this is not desirable, the **BREAK OFF** command may be executed in the program, in which case the FIREBERD will just set the **ERROR** status flag true upon detection of a program error; program execution will continue. This flag may be tested using the **IF [NOT] ERROR THEN (line number)** command, and should be reset using the **BREAK RESET** command (after any appropriate action has been taken). If the **ERROR** flag is set then the **BREAK WARN** command will print the error message "WARNING: Illegal input."; otherwise, it will perform no action.

See also: **IF-THEN**

EXAMPLE:

```
> 10 BREAK ON                    : this program runs normally
> 20 INPUT 2/" COMMAND"          : ask the user for a command to execute
> 30 & 2                         : then execute it
> RUN
COMMAND? BEEPP                   : if the user enters an illegal command

ERROR: Unrecognized command.     : then the program breaks
Break in line 30

> 10 BREAK OFF                   : this program runs in BREAK OFF mode
> 15 BREAK RESET                 : make sure the ERROR flag is reset
> 20 INPUT 2/"COMMAND"           : ask the user for a command to execute
> 30 & 2                         : then execute it
> 40 BREAK WARN                  : if there was an error then warn the user
> 50 IF ERROR THEN 15            :   and ask for the command again
> 60 END                         : else end the program
> RUN
COMMAND? BEEPP                   : now if the user enters and illegal command

WARNING: Illegal input           : the program issues a warning

COMMAND? BEEP                    : then asks for the command again until
                                 :   it is entered correctly
                                 : the FIREBERD then beeps
*DONE*
```

Asynchronous Character Format

CHA FOR	: print the character format status
CHA FOR DAT [5|6|7|8]	: select the number of data bits
CHA FOR PAR [EVE|ODD|NON]	: select the parity
CHA FOR STO [1|1.5|2]	: select the number of stop bits

This command selects or prints the asynchronous character format.

EXAMPLE:

```
> CHA FOR                 : print the current character format
>
DAT 7
PAR ODD
STO 2
> CHA FOR PAR EVE         : then select even parity
> CHA FOR PAR             : and print the newly selected parity
>
EVE
>
```

CHAIN (program name)

When this command is executed by a running program, the FIREBERD loads the named program from the ramdisk and starts executing it from its beginning. All program variables will be cleared prior to executing the newly loaded program unless you specify otherwise using the **COMMON** command. Note that when the new program is loaded, it overwrites the previous program.

See also: **COMMON**

EXAMPLE:

```
> 10 LPRINT "DAISY, GIVE ME YOUR ANSWER"
> 20 END                    : enter a program
> SAVE DAISY                : and save it under the name DAISY
> NEW                       : then prepare to enter a new program
> 10 LPRINT "I'M CHAINING TO DAISY!"
> 20 CHAIN DAISY            : and make it CHAIN to DAISY
> 30 LPRINT "I NEVER GET HERE!"
> 40 END
> RUN                       : then run the new program
>
I'M CHAINING TO DAISY!: (the new program is overwritten by DAISY)
DAISY, GIVE ME YOUR ANSWER

*Done*
> LIST
>

10 LPRINT "DAISY, GIVE ME YOUR ANSWER"
20 END

[xxx bytes free]
>
```

CLS

Clear the Terminal Screen

CLS

This command simply outputs 30 of the selected line terminator sequences (usually carriage return, linefeed for terminals) to the terminal. This has the effect of clearing the terminal screen of all previous outputs.

EXAMPLE:

```
> CLS                    : clear the terminal screen
>
```

COMMON

Issuing this command immediately prior to a **CHAIN** instruction will cause the program specified by the **CHAIN** command to inherit the values of all BERD-BASIC variables from the current working program. Otherwise, all variables will be cleared (to zero) before starting the execution of the **CHAIN**ed program.

See also: **CHAIN**

EXAMPLE:

```
> 10 LPRINT "AT MYPROG1, VARIABLE X IS STILL"
> 20 PRINT VAR X          : make program to print the value of variable X
> 30 CHAIN MYPROG2
> SAVE MYPROG1            : then save it under the name MYPROG1
> NEW                     : then prepare to enter another program
> 10 LPRINT "AT MYPROG2, VARIABLE X IS NOW"
> 20 PRINT VAR X          : that does the same thing
> 30 END
> SAVE MYPROG2            : and save it under the name MYPROG2
> NEW                     : then prepare to enter the main program
> 10 X=42                 : set variable X to 42
> 20 LPRINT "X IS INITIALLY"
> 30 PRINT VAR X          : then print out X's current value
> 40 COMMON               : then preserve this value thru next CHAIN
> 50 CHAIN MYPROG1        : and chain to MYPROG1
> RUN
>
X IS INITIALLY
42
AT MYPROG1, VARIABLE X IS STILL
42
AT MYPROG2, VARIABLE X IS NOW
0

*Done*
>
```

Continue Program Execution

CONT

This command causes the FIREBERD to continue executing a program from where it previously stopped, due to either the execution of a **STOP** command or the reception of a Control C from the controller. In this manner, the user can temporarily suspend program execution and resume it at a later time. Note that program execution cannot be **CONT**inued once any of the program lines have been changed.

See also: **STOP**

EXAMPLE:

```
> 10 LPRINT "FIRST HALF"      : that runs in two halves
> 20 STOP                     : with a STOP in the middle
> 30 LPRINT "SECOND HALF"
> 40 END
> RUN                         : then run the program
FIRST HALF                    : and the first half finishes

Break in Line 20

> CONT                        : then continue from after the STOP
SECOND HALF                   : and the program finishes

*Done*
>
```

DAT : print the currently selected data pattern
DAT (data) : select a new data pattern

This command prints or sets the current data pattern generated by the FIREBERD. The **(data)** pattern selected may be one of the following:

MAR	: all Mark
1:1	: alternating Mark and Space
63	: 63 bit pseudorandom
511	: 511 bit pseudorandom
2047	: 2047 bit pseudorandom
2 ^ 15-1	: 2^{15}-1 bit pseudorandom
2 ^ 20-1	: 2^{20}-1 bit pseudorandom
2 ^ 23-1	: 2^{23} bit pseudorandom
QRS	: QRSS pattern
PRG	: programmable pattern
FOX	: THE QUICK BROWN FOX...message
USE	: user-defined message

EXAMPLE:

```
> DAT                    : find the currently selected data pattern
>
63

> DAT PRG                : select the programmable pattern
>
```

Return the FIREBERD to its Power-up State

DEVICE CLEAR
BYE : short format for **DEVICE CLEAR** command

This command sets the FIREBERD back to its power-up state. It is typically used to "log out" at the end of a remote control session. This ensures that all temporary configurations (such as **PROMPT, ECHO, HOLD, ENTER,** etc.) are cleared, while leaving all NOVRAM configurations unchanged. In this way, the next user to "log in" to the FIREBERD is ensured a clean start-up with no possible surprise modes remaining.

EXAMPLE:

```
> BYE          : log-out command
               : note that there is no prompt returned.
```

DIR

This command causes the FIREBERD to print the names of all programs that are currently **SAVE**d in the ramdisk. Note that the ramdisk can only hold four programs at one time.

See also: **SAVE, LOAD, KILL**

EXAMPLE:

```
> 10 REM THIS IS
  PROGRAM LUPA          : enter a do-nothing program
> SAVE LUPA             : and save it under the name LUPA
> 10 REM THIS IS
  PROGRAM ABC           : then write another program (over the old one)
> SAVE ABC              : and save it under the name ABC
> DIR                   : list the names of the saved programs
>

"LUPA"      "ABC"
>
```

Display Hold

DIS HOL : print the current display hold status
DIS HOL [ON|OFF] : set the display hold mode

This command displays or sets the display hold mode. If the **DIS HOL ON** selection is chosen, the FIREBERD display buffers (accessed through the **ANA RES 1** and **ANA RES 2** commands) are frozen in their current state. Otherwise, with **DIS HOL OFF**, the display buffers are constantly updated to reflect the current FIREBERD results.

See also: **ANA RES 1, ANA RES 2**

EXAMPLE:

```
> DIS HOL                 : check the current display hold state
>
OFF

> DIS HOL ON              : then freeze the displays
> ANA RES 1 TIM           : and find the time at which they were frozen
>
15:30:19                  : (3:30 PM)
>
```

Front Panel Display Mode

DISPLAY : print the display status
DISPLAY [LOCAL|REMOTE] : set the front panel display mode

This command allows the remote controller to determine whether or not the front panel display switches will be active while the FIREBERD is under remote control. If the **DISPLAY LOCAL** selection is made, the four display switches will remain operative while the FIREBERD is under remote control (allowing users to scroll through all of the results). The **DISPLAY REMOTE** selection, on the other hand, disables the front panel display switches (along with all others) allowing only the remote controller to manipulate the displayed results (via the **ANA RES 1** and **ANA RES 2** commands).

See also: **ANA RES 1, ANA RES 2**

EXAMPLE:

```
> DISPLAY REMOTE        : set the display switches under remote control
> ANA RES 1 BIT ERR     : and display bit errors in the left display
>
12
> DISPLAY               : check the status of the DISPLAY command
>
REMOTE

> DISPLAY LOCAL         : then set the display under front panel control
>
```

Manipulate the DSR/DM Control Line

DSR	: print the status of the DSR/DM control line
DSR [ON|OFF]	: set the new state of the DSR/DM control line
DM ...	: alternate form of DSR ...

This command displays or sets the state of the DSR (for RS-232) or DM (for RS-449) interface control line. It is only valid when the FIREBERD is operating in DCE mode.

EXAMPLE:

```
> DSR                    : print the current state of the DSR line
>
OFF

> DSR ON                 : then turn it on
>
```

Manipulate the DTR/TR Control Line

DTR : print the status of the DTR/TR control line
DTR [ON|OFF] : set the new state of the DTR/TR control line

This command displays or sets the state of the DTR (for RS-232) or TR(for RS-449) interface control line. It is only valid when the FIREBERD is operating in DTE mode.

EXAMPLE:

```
> DTR                    : print the current state of the DTR line
>
OFF

> DTR ON                 : then turn it on
>
```

Select the Echo Mode

ECHO : print the echo function status
ECHO [OFF|ON] (option) : select the echo mode

This command determines whether the FIREBERD will echo characters received from the controller. If the **ECHO OFF** selection is chosen, no received characters will be echoed. Otherwise, with **ECHO ON**, all received characters will be echoed.

The **(option)** can be either **"NON-DESTRUCTIVE BACKSPACE"** or **"DESTRUCTIVE BACKSPACE"** (these can be abbreviated to **"N."** and **"D."** respectively). If the **NON-DESTRUCTIVE BACKSPACE** option is chosen, received (backspace) characters will be echoed unchanged. Many terminals will not, however, remove characters from the screen as they are backspaced over. This can result in confusing characters on both sides of the cursor following a backspace sequence. For this reason, the FIREBERD defaults to **DESTRUCTIVE BACKSPACE** mode in which it echoes the sequence (backspace), (space), (backspace) for every received (backspace) character. This has the effect of ensuring that all characters backspaced over are physically blanked from the terminal screen.

EXAMPLE:

```
> ECHO                    : print the ECHO status
>
ON DESTRUCTIVE BACKSPACE

> ECHO OFF                : then turn off the echo
>>                        : (assume we type ECHO ON - it's not echoed)
DESTRUCTIVE
  BACKSPACE               : (this is the status of ECHO ON)

>
```

End an ENTER Sequence or End a BERD-BASIC Program

END

The **END** command can have two different meanings to the FIREBERD. If it is issued while an **ENTER** sequence is in progress, it is interpreted as an "END of ENTER Sequence" command. Otherwise, if no **ENTER** sequence is in progress, it is interpreted as an "END of BERD-BASIC Program" command.

The "END of ENTER Sequence" command causes all commands entered during the sequence to be executed and any resultant contention error messages to be printed out. Note that the "END of ENTER Sequence" command always results in a test restart.

The "END of BERD-BASIC Program" command terminates program execution (returning the FIREBERD to **remote** mode) and causes the message "*Done*" to be sent to the controller, indicating that the program has terminated normally (as opposed to terminating because of an error or a STOP/Control C). If a program reaches its last line and attempts to execute further, an implied "END of BERD-BASIC Program" is executed (this means that a program need not have an **END** statement as its last instruction).

See also: **ENTER**

EXAMPLE:

```
> 10 ENTER                    : start an ENTER sequence
> 20 REM ENTER COMMANDS HERE
> 30 END                      : then end the ENTER sequence, not the program
> 40 LPRINT "I'M STILL RUNNING"
> 50 END                      : then end the program
> 60 LPRINT "I NEVER GET HERE"
> RUN                         : run the program
>
I'M STILL RUNNING

*Done*

> ENTER                       : start another ENTER sequence
< REM ENTER
  COMMANDS HERE               : and enter any desired commands
< END                         : then end the ENTER sequence
>                             : (and the prompt goes back to normal)
```

ENTER

This command allows all subsequent commands to be entered regardless of whether or not they generate a contention. This continues until an END command is received at which time all of the previous commands are executed and any pending contention error messages are printed. This allows the controller to enter a series of commands in any order without worrying if there might be a temporary contention generated because of an intermediate set-up. Note that during an **ENTER** sequence, the prompt (if enabled) will change from the normal ">" to a "<" as a reminder that the commands being entered are not yet being executed or checked for contentions.

See also: **END**

EXAMPLE:

```
> TIM MOD ASY          : select async timing mode
> ENTER                : then start the ENTER sequence
< GEN CLK SYN          : select the synthesized clock
< SYN FRE 1000         : and set it to 1MHz (a contention with async)
< REM NOTICE NO CONTENTION ERROR MESSAGE WAS GENERATED
< TIM MOD SYN          : then select sync timing mode
< END                  : and end the ENTER sequence
>
```

ERR INS	: print the error insertion status
ERR INS [OFF|10^-3]	: select the constant error rate
ERR INS SIN	: insert a single error
ERROR	: alternate form for **ERR INS SIN**

This command sets or displays the error insertion status. Selecting **ERR INS 10^-3** causes one error to be injected into the generated data for every 1000 bits. Selecting **ERR INS OFF** terminates this condition. The **ERR INS SIN** command causes a single error to be inserted into the generated data. Note that a single error insertion will terminate a 10^-3 error insertion rate.

EXAMPLE:

```
> ERR INS OFF          : turn off the automatic error insertion
> ERR INS              : print the error insertion status
>
OFF

> ERR INS 10^-3        : enable automatic error insertion
> ERROR                : then insert a single error
> ERR INS SIN          : and another
> ERR INS              : then check the error insertion status
>
OFF                    : (notice that the ERROR command turned it OFF)
>
```

Generator Clock Select

GEN CLK	; print the generator clock status
GEN CLK [SYN|INT|BNC]	; select the source of the generator clock

This command selects from where the FIREBERD will get its generator clock. Either the **SYN**thesizer, **INT**erface, or rear panel **BNC** connector may be specified.

EXAMPLE:

```
> GEN CLK SYN          : select the synthesizer as generator clock
> SYN FRE 9.6          : and set the synthesizer frequency to 9.6 kHz
> GEN CLK              : print the generator clock selection
>
SYN
>
```

GOSUB (line number)

When this command is executed by a running program, the FIREBERD will start executing commands from the subroutine beginning at the specified line number until a **RETURN** statement is encountered. Following the execution of the **RETURN** statement, the FIREBERD will resume program execution from the next command following the **GOSUB**. Note that a subroutine may, in fact, call other subroutines up to 10 levels deep (that is, there may be up to 10 **GOSUB**s pending **RETURN**s).

See also: **RETURN**

EXAMPLE:

```
> 10 LPRINT "CALLING LINE 90"   : make a program to
> 20 GOSUB 90                   : execute the subroutine starting at line 90
> 30 LPRINT "I'M BACK"
> 40 END                        : then end
> 90 REM THIS IS IT!            : the subroutine prints its message and
> 100 LPRINT "I'M IN THE SUBROUTINE!"
> 110 RETURN                    : then returns back to the main program
> RUN
CALLING LINE 90
I'M IN THE SUBROUTINE!
I'M BACK

*Done*
>
```

Transfer Execution to Specified Line

GOTO (line number)

When this command is encountered in a running program, it causes the FIREBERD to start executing commands from the specified line (rather than from the line following the **GOTO** command). This allows you to alter the sequence in which the FIREBERD is to execute the commands in a program.

See also: **IF-THEN**

EXAMPLE:

```
> 10 LPRINT "HELLO
  THERE!"                  : just print any message
> 20 GOTO 10               : then go back and do it again (forever)
> RUN
HELLO THERE!
HELLO THERE!
HELLO THERE!
HELLO THERE!
HEL\                       : (stop the program by sending a Control C)

Break in Line 20           : (and the FIREBERD will stop)

>
```

HELLO
AUX 40 : alternate form of HELLO command.

This command causes the FIREBERD to print the revision level of its software along with the TTC logo.

EXAMPLE:

```
> HELLO                    : print the software revision level
>

... (TTC's logo printed here)
   .
   .
   .

Fireberd 6000, Software Version "B", (c) TTC, 3 SEP 1986
>
```

HELP!	: display list of all commands
HELP	: display the main help screen
HELP (command name)	: display the parameter syntax for a command
HELP AUX	: print the list of AUX functions
HELP PAGE	: print the list of available help pages

This command provides the remote user with access to all of the FIREBERD on-line help functions. The main help screen shows how to get more detailed help on any particular function of remote control. Entering the **HELP (command name)** option will cause the FIREBERD to both print a list of all possible parameters associated with the specified command, and set the help/status macro (which is recalled using a "*" in the command line) equal to the specified command. This allows the user a short-cut reference (using "*") to the command for which help was just requested. See the example.

See also: **PAGE**

EXAMPLE:

```
> HELP TIM MOD          : print list of parameters for the TIM MOD
                          command
>
ASY
SYN
REC

> * SYN                 : then select the sync choice (TIM MOD SYN)
> HELP                  : then print the main help screen
>
  .
  .                     : (the help screen is printed)
  .
>
```

HOLD

This command causes the FIREBERD to temporarily hold all printer outputs (in the print buffer) until the reception of a **REL**ease command. Note that while a printer **HOLD** is in progress, the prompt character (if enabled) will change from the normal ">" to a "+" to warn the user that printouts are being held.

See also: **REL**

EXAMPLE:

```
> HOLD                  : hold all printouts for now
+ TIME                  : then print the time and date
+ DATE                  : (note that nothing gets printed yet)
+ REL                   : then release the printer hold
>
12:30:11                : and the FIREBERD prints the time

10 APR 86               : and date
>
```

GOTO Specified Line If Condition Is True

IF [NOT] (status) THEN (line number) : test a status
IF (macro)(eq relation)(string) $ THEN (line number) : test a macro value
IF (value)(relation)(value) THEN (line number) : test a result

This command enables a program to change the sequence in which it executes depending on the value of one of the FIREBERD status flags, results, user macros or program variables. If the specified condition is met then the FIREBERD will perform a **GOTO** operation to the indicated **(line number)**, causing all subsequent commands to be executed from there. Otherwise, this command has no effect and program execution continues with the next command in sequence.

The first format of the command is used to test the state of one of the FIREBERD status flags. The **(status)** field may either be **ERROR** (indicating that a program error has occurred in **BREAK OFF** mode) or the name of one of the FIREBERD front panel status LEDS. The front panel status LED names are as follows:

ALM 1	**ALM 2**	**CLK INV**	**CODE**	**CTS**
DATA INV	**DM**	**DSR**	**DTR**	**FRM SYNC**
LL	**NO CLK**	**RLSD**	**RL**	**RR**
RTS	**TM**	**SYNC LOST**	**SYNC**	

The command **IF (status) THEN (line number)** will cause the program execution to jump to the specified line only if the specified status is true (LED is on). Similarly, the command **IF NOT (status) THEN (line number)** will cause the program execution to jump to the specified line only if the status is false (LED is off).

The second format of the command is used to test the contents of macro **# (macro)**; where **(macro)** is a the user macro number from 0 to 9. The program execution will jump to the specified line if the **(eq relation)** exists between user macro number **(macro)** and the **(string)**; where **(eq relation)** may be either "=" (equal) or "<>" (not-equal).

The third format of the command is used to test the value of one of the FIREBERD results. The **(value)** field may be either the name of any printable result (except **JTR FRE** and **RCV COD** -- See the **PRINT** command for a complete list of printable results) or a literal numeric constant (in fixed decimal format or scientific notation). Several example **(value)** fields follow:

BIT ERR : number of bit errors since test start
X : the value of program variable X
12.5 : the value 12.5
1.56E-3 : the value 0.00156 in scientific notation

Note: When using the current **TIM**e or **DAT**e results, only the hours:minutes (**hh:mm**) or the date (**dd**) are evaluated for comparison, respectively.

The **(relation)** field denotes a BASIC relational operator. These includes

<>	: not equal to
=	: equal to
<	: less than
>	: greater than
<=	: less than or equal to
>=	: greater than or equal to

The program will jump to the specified **(line number)** if the specified **(relation)** exists between the two **(value)**s.

See also: **PRINT, GOTO, BREAK**

EXAMPLE:

```
> 5 IF TIM <> 23:00 THEN: wait til 11:00 PM
5
> 10 RES                    : then restart the test
> 20 WAIT SYNC              : then wait for sync to be acquired
> 30 WAIT 10                : then accumulate results for 10 seconds
> 40 IF AVG BER >
    1E-3 THEN 70            : check if the average BER is greater than .001
> 50 LPRINT "BER IS OK"     : say that its OK if not
> 60 END                    : then end the program
> 70 LPRINT "BER IS HIGH"   : otherwise say that the BER is too high
> 580 END                   : then end the program
> RUN                       : then run the program
BER IS OK                   : (assuming BER was less than .001)

*Done*

> NEW
> 10 INT SET INT 232 EMU DCE
> 20 IF RTS THEN 60                 : check if we are receiving RTS true
> 30 INT SET INT 232 CTS OFF        : turn CTS off if not
> 40 GOTO 20                        :
> 60 INT SET INT 232 CTS ON         : else turn CTS on
> 70 GOTO 20                        : then wait for more changes
> RUN

/                                   : (press Control C to stop the program)
Break in line 20

> NEW
> 10 INPUT 2/"YES OR NO"            : ask the user to enter yes or no into
                                      macro 2
```

```
> 20 IF 2=YES$ THEN 60          : if the answer was yes then skip ahead
> 30 IF 2<>NO$ THEN 10          : else if the answer was not no then
                                  ask again
> 40 LPRINT "YOU SAID NO!"      : else tell the user what he said
> 50 GOTO 70
> 60 LPRINT "YOU SAID YES!"
> 70 END
> RUN
YES OR NO? YE
YES OR NO? YES
YOU SAID YES!

*Done*
```

Enter User Input into a Macro

INPUT (macro) : Enter input from terminal into macro
INPUT (macro)/(string) : prompt user with string, then input into macro

This command is used by a running program to receive input from the user. The FIREBERD will first prompt the user with the **(string)**, if specified, followed by a question-mark. It will then wait for the user to enter a line of characters followed by a statement terminator. That line (up to 31 characters) will then be stored as macro **# (macro)** and program execution will continue.
will continue.

See also: **MACRO**

EXAMPLE:

```
> 10 BREAK OFF              : this program runs in BREAK OFF mode
> 15 BREAK RESET            : make sure the ERROR flag is reset
> 20 INPUT 1/"WHAT
    TIME IS IT"             : then ask the user what time it is
> 30 TIME & 1               : then set the time to what he said
> 40 BREAK WARN             : if there was an error then print a warning
> 50 IF ERROR THEN 15       :  and go back and ask again
> 60 END                    : else end the program
> RUN                       : run the program to set the time
WHAT TIME IS IT? NOON       : enter an illegal value for the time

WARNING: Illegal input.

WHAT TIME IS IT? 12:00      : then enter a legal value

*Done*

>
```

INT SET	: print the current interface setup
INT SET INT 232 (config)	: select the internal RS-232 interface
INT SET DS0 (config)	: select the DS0/DS0A interface
INT SET DS1/FE (config)	: select the DS1/T1/Fe interface
INT SET DS1/T1 (config)	: select the DS1/T1 interface
INT SET 8M G703 (config)	: select the 8Mb G.703 interface
INT SET 64 G703 (config)	: select the 64 kb/s G.703 interface
INT SET 2M G704 (config)	: select the 2048 kb/s G.703 interface
INT SET EXT	: select the generic "external" interface

This command selects the interface to be used by the FIREBERD. For interfaces that are remotely controllable, it also configures the interface. The **INT SET EXT** command is used to select one of the non-remote controllable FIREBERD 2000 data interfaces. The **INT SET INT 232 (config)** command is used to select the internal RS-232 interface for use.

Selecting one of the FIREBERD 6000 remote controllable interfaces allows the specification of a **(config)** field to configure the interface. the **(config)** field for each of these interfaces is as follows:

INT 232 (config)

EMU [DTE|DCE]	: emulate DTE or DCE
RI [ON|OFF]	: control RI signal
CTS [ON|OFF|AUT]	: control CTS signal

DS0 (config)

TES MOD [DS0|DS0A]	: interface test mode
LOO COD [COD|TES]	: loop code
DAT [L NEA|L FAR|BIP]	: data type
LOO [DSU|CHA|OCU|NON]	: loop back
RAT [2.4K|4.8K|9.6K|56K]	: data rate

DS1/FE (config)

RES [STD|LIV]	: result mode
THR [ON|OFF]	: thru data mode
INP [TER|BRI|DSX MON]	: input level
COD [AMI|B8ZS]	: line code
FRA [OFF|D4|FE]	: framing format
LBO [0|-7.5|15]	: line buildout (db)
BPV INS [OFF|SIN|RAT (rate)]	: bpv insertion. (rate is in the form "xE-y" where 0<=x<=9 and 3<=y<=9.

DS1/T1 (config)

RES [STD|LIV]	: result mode
THR [TLB|LLB|OFF]	: thru data mode
INP [TER|BRI|DSX MON]	: input level
COD [AMI|B8ZS]	: line code
FRA [OFF|D4|FE|SLC]	: framing format
LBO [0|-7.5|-15]	: line buildout (db)
ERR INS [OFF|BPV|LOG| L+BPV|SIN|RAT (rate)]	: bpv insertion. (rate is in the form "xE-y" where 0<=x<=9 and 3<=y<=9.

8 M G703 (config)

TIM [XTA|REC]	: timing
MOD [NOR|BPV]	: result mode
IMP [TER|BRI]	: input impedance

2MG704 (config)

FRA UNFRA	: select Unframed mode
FRA FRA [CRC4 (OFF, ON)| TS16 (OFF, ON)]	: select Framed mode with multiframe options
RES [STD|LIV]	: set results criteria
TLB [OFF|ON]	: set thru-loopback
ERR INS [OFF|SIN|RAT]	: set error insert rate
LIN [TER|BRI]	: select input impedance
COD [AMI|HDB]	: select code standard
CLK [XTA|REC]	: select timing source

64G703 (config)

MOD [COD|CEN|CON DTE/ CON DCE]	: select interface mode
OCT [ON|OFF]	: set octet timing status
INP [BRI|TER]	: set input impedance
ERR INS [OFF|SIN|RAT]	: set error insert rate
THR [OFF|ON]	: select thru data mode

For more information on the operation of any of the above interfaces, please see the manual section for the interface itself.

See also: **AUX 5**

INT SET

EXAMPLE:

```
> INT SET                    : check the current interface setup
>
INT 232 EMU DTE              : (internal RS-232 in DTE mode)

> INT SET INT 232 EMU DCE: then switch over to DCE mode
> INT SET INT 232 RI ON    : then turn the RI control line on
> INT SET INT 232          : get the status of the internal 232 intf
>
EMU DCE
RI ON
CTS OFF

>
```

ISU	: Display the currently active interface
ISU [1|2|3|4]	: Activate interface in single chassis configuration
ISU [1|2:1|2|3|4]	: Activate interface in dual chassis configuration

The **ISU** command controls the selection of interfaces from a remote control unit. **ISU** displays the currently selected interface within the ISU-6000. **ISU slot-number** activates a new interface in a single chassis configuration. **ISU chassis-number:slot number** activates a new interface in a dual chassis configuration.

When changing the currently active interface, the indicator light associated with the selected interface (to the left of the slot number on the ISU-6000) is illuminated. If none of the interfaces in the ISU-6000 are active, the following error message is displayed:

ISU IS NOT SELECTED

EXAMPLE 1:

> **ISU**	: display the currently active interface
V.35	: the interface in slot #3 is selected
>**ISU 3**	

EXAMPLE 2:

>**ISU 2:3**	: the interface in chassis 2, slot #3 is selected.

JIT	: print current jitter status
JIT MAS (mask)	: select jitter mask
JIT WDB [FUL|>8 KHZ]	: select wide-band bandwidth
JIT WDB (threshold)	: set wide-band hit threshold
JIT S/A SWE [ON|OFF]	: set spectrum analyzer sweep
JIT S/A FRE (sa freq)	: set spectrum analyzer frequency
JIT GEN MOD (mod)	: set jitter generator modulation
JIT GEN FRE [(gen fre)|UP| DOW|SWE]	: set jitter generator frequency
JIT GEN AMP [(amplitude)|MAS| XFE|UP|DOW]	: set jitter generator amplitude
JIT LIS	: list spectrum analyzer results
JIT GRA	: graph spectrum analyzer results

This command manipulates the jitter parameters for the DS1 wide-band jitter, jitter spectrum analyzer, and jitter generator options.

The **JIT MAS (mask)** command can be used to select the jitter mask. The **(mask)** may be any one of the following: O.171, 41451, 43801, 62411-1983, 62411-1985.

The wide-band jitter circuit may be set to measure either full-bandwidth jitter or jitter above 8 kHz using the **JIT WDB FUL** or **JIT WDB>8KHZ** commands respectively. The jitter hit **(threshold)** is a number in the form "n.n", ranging from 0.0 to 6.5 unit intervals.

The spectrum analyzer may be set to either monitor a single frequency or sweep all 40 frequencies using the **JIT S/A SWE OFF** or **JIT S/A SWE ON** commands, respectively. When using **JIT S/A SWE OFF**, the frequency to monitor is set using the **JIT S/A FRE (sa freq)** command; where **(sa freq)** is one of the following frequencies (in hz):

10, 20, 30, 40, 50, 60, 70, 80
100, 133, 166, 200, 233, 300, 400, 500, 600, 700, 800
1000, 1330, 1660, 2000, 2330, 3000, 4000, 5000, 6000, 7000, 8000
10000, 13300, 16600, 20000, 23300, 26600, 30000, 33300, 36600, or **40000**

The jitter generator modulation is controlled with the **JIT GEN MOD (mod)** command; where **(mod)** is one of the following modulation waveforms:

OFF	: no modulation takes place
SIN	: sine wave
SQU	: square wave

TRI	: triangle wave
RAM	: ramp wave
EXT	: external waveform

The jitter generator frequency may be set to sweep the same 40 frequencies as the spectrum analyzer by the **JIT GEN FRE SWE** command. If the generator and the spectrum analyzer are both sweeping, then the sweep will be synchronized. The generator frequency may also be set to a fixed frequency by the **JIT GEN FRE (gen fre)**. The **(gen fre)** must be between 10 Hz and 40 kHz with not more than 4 significant digits. The set value of **(gen fre)** may also be increased or decreased by the **JIT GEN FRE UP** and **JIT GEN FRE DOW** commands, respectively. These commands will increase or decrease the frequency to the next higher or lower frequency from the list of 40 sweep frequencies.

The jitter generator amplitude may be set to a given amplitude by the **JIT GEN AMP (amplitude)** command. The **(amplitude)** must be in the range of 0.00 and 13.00 unit intervals in .02 unit interval increments. The **JIT GEN AMP UP** and **JIT GEN AMP DOW** commands can be used to increase or decrease the amplitude by 0.02 unit intervals, respectively. The amplitude may also be set to the selected mask (see **JIT MAS (mask)**) by the **JIT GEN AMP MAS** command. A transfer function can also be selected by the **JIT GEN AMP XFE** command. This command will set the amplitude to be half of the selected mask.

The **JIT LIS** command is used to initiate a jitter spectrum analyzer frequency list printout; likewise, the **JIT GRA** command is used to initiate a jitter spectrum analyzer frequency graph.

EXAMPLE:

```
> JIT MAS 0.171          : select the 0.171 jitter mask
> JIT WDB FULL           : do full bandwidth ccitt measurements
> JIT WDB 4.2            : set the jitter hit level at 4.2 ui
> JIT S/A SWE ON         : set the spectrum analyzer to sweep all freqs
> JIT GEN AMP 3.5        : set the jitter generator amplitude
> JIT GEN MOD OFF        : but turn off jitter generation (i.e. 0.0 UI)
> JIT                    : print the current jitter setup
>
MAS    0.171
WDB    FUL
       THR 4.2
S/A    SWE ON
       FRE 10 (Hz)
GEN    MOD OFF
       FRE 10 (Hz)
       AMP 3.5 (UI) selected, actual level generated = 0.0 (UI)

>
```

JIT	:display current jitter status
JIT MAS (mask)	:select jitter mask
JIT S/A SWE [ON|OFF]	:set spectrum analyzer sweep
JIT S/A FRE (sa frequency)	:set spectrum analyzer frequency
JIT RAN [BAND1|BAND2]	:select measurement range
JIT GEN MOD (mod)	:select modulation waveform
JIT GEN FRE [(frequency)]|UP|DOW|SWE]	:set jitter waveform frequency
JIT GEN AMP [(amplitude)|MAS|XFE|UP|DOWN]	:set jitter waveform amplitude
JIT HIT [(hit threshold)|UP|DOW]	:set jitter hit threshold
JIT LIS	:list spectrum analyzer results
JIT GRA	:graph spectrum analyzer results

The **JIT** command controls the G.703 jitter parameters. The **JIT MAS (mask)** command can be used to select the jitter mask. The **(mask)** may be any one of the following: **O.171, PTT, G.823, or G.823N.**

The spectrum analyzer may be set to monitor a single frequency or sweep all frequencies using **JIT S/A SWE OFF** or **JIT S/A SWE ON,** respectively. When selecting **JIT S/A SWE OFF**, the frequency to be monitored is specified using the **JIT S/A FRE (sa freq)** command. The following frequencies may be specified when using the 64 kb/s G.703 Data Interface (in Hz):

10, 20, 30, 40, 50, 60, 70, 80 100, 133, 166, 200, 233, 300, 400, 500, 600, 700, 800
1000, 1330, 1660, 2000, 2330, 3000, 4000, 5000, 6000, 7000, 8000
10000, 13300, 16600, 20000, 23300, 26600, 30000

The following frequencies may be specified when using the 2048 kb/s G.704 Framing Interface (in Hz):

10, 20, 30, 40, 50, 60, 70, 80 100, 133, 166, 200, 233, 300, 400, 500, 600, 700, 800
1000, 1330, 1660, 2000, 2330, 3000, 4000, 5000, 6000, 7000, 8000
10000, 13300, 16600, 20000, 23300, 26600, 30000 33000, 36600, 40000, 43300, 46600, 50000, 53300, 56600, 60000 63300, 66000, 70000, 73300, 76600, 80000, 83300, 86600, 90000 93300, 96600
100000

The **JIT RAN** command specifies the jitter measurement range based on the BAND selected (BAND 1 or BAND 2) and the interface installed (64 kb/s or 2048 kb/s). See Section 9 of the FIREBERD 6000 Communications Analyzer Operating Manual for a list of valid jitter measurement ranges.

The **JIT GEN MOD** command controls jitter generator modulation. The (mod) parameter may be any of the following:

OFF	:no modulation takes place
SIN	:sine wave
SQU	:square wave
TRI	:triangle wave
RAM	:ramp wave
EXT1	:external wave using generator band 1
EXT2	:external wave using generator band 2

The jitter generator frequency may be set to sweep the same frequencies as the spectrum analyzer using the **JIT GEN FRE SWE** command. If the generator and the spectrum analyzer are both sweeping, then the sweep is synchronized. The generator frequency may also be set to a fixed frequency using the **JIT GEN FRE (frequency)** command. The **JIT GEN FRE UP** and **JIT GEN FRE DOWN** commands can be used to set the frequency to its next highest or lowest value, respectively (see the list provided in the spectrum analyzer description).

The jitter generator amplitude may be set to a given amplitude by the **JIT GEN AMP (amplitude)** command. The **(amplitude)** must be in the range of 0.00 and 12.00 unit UI in .02 UI increments when using the 2048 kb/s G.704 Framing Interface, and 0.00 and 1.500 UI in .005 increments when using the 64 kb/s G.703 Data Interface. The **JIT GEN AMP UP** and **JIT GEN AMP DOW** commands increase or decrease the amplitude by 0.02 or 0.05 UI, respectively. The amplitude may also be set to the select mask (see JIT MAS (mask)) by the **JIT GEN AMP MAS** command. A transfer function can also be selected by the **JIT GEN AMP XFE** command. This command sets the amplitude to a value of half of the selected mask at the current frequency.

The **JIT HIT** command controls the jitter hit threshold. The **JIT HIT (hit threshold)** command allows a jitter hit threshold value to be set based on the band (BAND 1 or BAND 2) and the interface installed (64 kb/s or 2048 kb/s), as follows:

Interface	Band	Range	Increments
2048 kb/s	Band 1	0.1—6.0 Ulp	.1 Ulp
	Band 2	0.05—0.60 Ulp	.05 Ulp
64 kb/s	Band 1	0.05—0.75 Ulp	.05 Ulp
	Band 2	0.05—0.15 Ulp	.05 Ulp

EXAMPLE:

> **JIT MAS G.823**	:set the G.823 jitter mask
> **JIT S/A SWE ON**	:set the spectrum analyzer to sweep all frequencies.
> **JIT GEN MOD SIN**	:set sine wave
> **JIT GEN AMP 0.10**	:set the jitter generator amplitude
(64 kb/s int.)	
> **JIT HIT 0.10**	:set jitter hit threshold

Delete a Saved Program

KILL (program name)

This command removes (unsaves) the previously **SAVE**d program from the ramdisk. Another program may now be **SAVE**d in its place. Note that using this command is the only way to reduce the number of programs saved in the ramdisk, thus freeing room for new programs.

See also: **DIR, LOAD, SAVE**

EXAMPLE:

```
> 10 REM THIS
    IS A DUMMY          : make up a dummy program that we can SAVE
> SAVE DUMMY 1          : and save it under the name DUMMY 1
> SAVE DUMMY 2          : as well as the name DUMMY 2 and DUMMY 3
> SAVE DUMMY 3          : (note that we now have three copies saved)
> DIR                   : then get a directory of what's stored
>

"DUMMY 1"     "DUMMY 2"     "DUMMY 3"

> KILL DUMMY 2          : then delete the copy named DUMMY 2
> DIR                   : and see what's out there now
>

"DUMMY 1"     "DUMMY 3"

>
```

Print the State of Various Front Panel LEDs

LEDS

This command causes the FIREBERD to print the state of various front panel warning and status LEDs. They include:

SYNC LOST
IN SYNC
AUX FUNCTION IN USE
NO GENERATOR CLOCK
RCV DATA INVERT
RCV CLOCK INVERT
NO RECEIVER CLOCK
T-CARRIER MODE
DTE MODE
DCE MODE
OCTET TIMING DETECTED (64 kb/s G.703 interface only)

EXAMPLE:

```
> LEDS                    : check which LEDs are active
>
IN SYNC
DATA GENERATOR ACTIVE
DTE MODE

>
```

Assign A New Value To A Variable

LET(variable)=(expression) : variable assignment using optional "LET"
(variable)=(expression) : variable assignment without "LET"

This command assigns the value of **(expression)** to the specified variable. The **(variable)** name must be a single letter from "A" to "Z". Program variables can only accommodate integers in the range of 0 to 32767. The **(expression)** may contain one or more integers or **(variables)**s added together or subtracted from each other. Note that the word **"LET"** is optional in all variable assignments.

EXAMPLE:

```
> 10 LET B=2                    : set variable B to 2
> 20 PRINT  B                   : then print the contents of variable B
> 30 LET B=B+B-1                : then calculate a new value of B
> 40 C=B                        : then set variable C the same as B
> 50 IF  C < 10 THEN 20         : and if the result is less than 10 then go back
> 70 PRINT B/
   "VARIABLE B IS"              : else print the final value
> RUN                           : and run the program
 2
 3
 5
 9
VARIABLE B IS
 17

*Done*
>
```

List the Working BERD-BASIC Program

LIST : list entire program
LIST (line number) : list, starting with specified line

This commands lists, with lines in ascending numerical order, the contents of the working program. It also prints the number of unused bytes in program memory (where one byte is approximately equal to one character of program). If the **(line number)** is not specified, listing will begin with the first line of the program. Otherwise, the listing will start with the specified line. Listing always ends at the last line of the program.

EXAMPLE:

```
> 10 BEEP                  : enter BEEP as line 10
> 30 PRINT BER             : enter PRINT BER as line 30
> 20 CLS                   : enter CLS as line 20
> LIST                     : then list the current program
>

10 BEEP
20 CLS                     : then list the working program
30 PRINT BER

[xxx bytes free]
>
```

Load a Saved Program

LOAD (program name)

This command retrieves a copy of a previously **SAVE**d program from the ramdisk, making it the working program. Note that when a new program is loaded, it overwrites the previous working program.

See also: **DIR, KILL, SAVE**

EXAMPLE:

```
> 10 REM THIS PROG IS SAVED   : make a program to save in permanent
                                storage
> SAVE DOG                    : and save it under the name DOG
> NEW                         : then wipe out the working program
> 10 REM GARBAGE              : and enter a meaningless new one
> LOAD DOG                    : then load the original prog back
> LIST                        : and check to see if it is there
>

10 REM THIS PROG IS SAVED

[xxx bytes free]
>
```

Return the FIREBERD to Local Mode

LOCAL
/ : alternate short form of **LOCAL** command

This command causes the FIREBERD to return to local mode, making the front panel switches again active. The FIREBERD will remain in local mode until it receives a valid remote control command from the controller. Note that this command may be abbreviated by a "/".

EXAMPLE:

```
> BEEP                    : beep and enter remote control mode
> LOCAL                   : and then return to local (front panel) mode
> REM LETS GO REMOTE!     : then go back to remote mode
> /                       : and quickly back to local mode
>
```

Generate Loop Codes

LOO [UP| DOW] : generate loop up or loop down codes

This command generates **LOO**p **UP** or **LOO**p **DOW**n codes from a T-carrier interface. Note that it is only valid when operating a T-carrier interface not in live mode.

EXAMPLE:

```
> LOO UP                          : generate a loop up code
Transmitting loop code. Please wait ...
Loop code transmission successful.

> LOO DOW                         : generate a loop down code
Transmitting loop code. Please wait ...
30 second timeout exceeded.

>
```

Print the Literal Text String

LPRINT (string)

This command causes the **(string)** to be sent to the printer. Specifying **LPRINT** without any string just causes a blank line to be printed. Note that, like all **strings**, if there are any imbedded spaces or special characters in it, it must be surrounded by double quotes Otherwise, the quotes are optional.

EXAMPLE:

```
> LPRINT HELLO              : print the word HELLO
>
HELLO

> LPRINT "HELLO THERE"      : then print the phrase HELLO THERE
>
HELLO THERE

>
```

Set or Display User Macros

MACRO	: print all currently defined macros
MACRO (number)	: print the definition of macro (number)
MACRO (number)/	: clear macro (number)
MACRO (number)/(string)	: define macro (number) as (string)

Using this command, the controller may display or define macros to be used as an aid in entering commands. A macro is a (possibly long) string of characters that the user can define once and recall easily many times using just one or two keystrokes. They are typically used if the user knows in advance that the same command (or part of a command) may need to be entered many times. The macro **(number)** may range from 0 to 9 and the macro **(string)** may be from 1 to 31 characters in length. The user-defined macros are recalled by typing **"&(number)"** in the command line whenever the macro substitution is to take place.

EXAMPLE:

```
> MACRO 1/TIME                    : define macro 1 as TIME
> MACRO 2/"PRINT BIT ERR"         : and define macro 2 as PRINT BIT ERR
> &1                              : then print the current time of
                                    day (TIME)
>
20:07:13

> &1 21:08                        : then set the time ahead one hour
                                    (TIME 21:08)
> &2                              : then print the bit error count
                                    (PRINT BIT ERR)
>
10093
> MACRO 2                         : print the contents of macro 2
>
PRINT BIT ERR

>
```

NEW

This command causes the FIREBERD to clear the working program memory. It should be used prior to entering a new program to ensure that no lines are left over from a previous program. This leaves the user with 510 bytes (each good for approximately one stored character) of memory free for the new program.

EXAMPLE:

```
> 10 REM THIS IS LEFTOVER   : enter something to be wiped out by the
> 20 LPRINT "SO IS THIS"    : NEW command
> NEW                       : then wipe out the entire program
> LIST                      : and make sure there is nothing left

[510 bytes free]

>
```

NUMBER – list all of the command numbers
NUMBER (command) – list the parameter numbers for a command

This command lists the internal command or parameter numbers assigned to each command or command parameter. As a shortcut to entering the normal command or parameter name, the user may enter a pound sign **(#)** followed by the command or parameter number. A slash character **("/")** may optionally separate this command or parameter number from any following parameter selections. Note that the "/" character is only required when following a numbered command or parameter selection by a numeric parameter (that is, a parameter that is not numbered using the **"#"** yet is numeric anyway – the slash is used to separate the two side-by-side numbers). Note that when entering an **AUX** function using command numbers, both the **"AUX"** and the auxiliary function numbers are treated separately. For example, the command number for **AUX** is **09**; to reference AUX function 30, you would enter either **#9#30** or **#9/30**. The internal command and parameter numbers may change in any future software enhancements.

EXAMPLE: For this example, assume that the command number for the **CHA FOR** command is **13** and that its parameters are numbered as follows:

(01) DAT	**(01) 5**
	(02) 6
	(03) 7
	(04) 8
(02) PAR	**(01) ODD**
	(02) EVE
	(03) NON
(03) STO	**(01) 1.5**
	(02) 1
	(03) 2

> #13 #1 #2	: select **CHA FOR DAT 6**
> #13 #1/6	: also select **CHA FOR DAT 6**
> #13 /DAT 6	: also select **CHA FOR DAT 6**
> #13 #2 NON	: select **CHA FOR PAR NON**
> #13 #2/NON	: also select **CHA FOR PAR NON**
> CHA FOR #2 #3	: also select **CHA FOR PAR NON**

```
> #13#2                    : print the selected parity (CHA FOR PAR)
>
NON

> #13                      : print the selected character format (CHA FOR)
>
DAT 6
PAR NON
STO 1.5
>
```

Transmit Octet Timing for 64kb G.703 Interface

OCT TIM : print the octet timing status
OCT TIM [ON|OFF] : enable or disable octet timing transmission

This command is only valid when using the 64kb G.703 interface. When **OCT TIM ON** is selected, the FIREBERD will transmit octet timing. Otherwise, with **OCT TIM OFF** selected, the FIREBERD will not transmit octet timing.

EXAMPLE:

> OCT TIM : check the current octet timing status
>
OFF : (we are not currently transmitting it)

> OCT TIM ON : start transmitting octet timing
>

OPTIONS

This command lists the options currently installed in the FIREBERD mainframe. The options include:

WIDE BAND JITTER
JITTER SPECTRUM ANALYZER
JITTER GENERATION
IEEE-488 PRINTER/CONTROLLER
DIGITAL CLOCK RECOVERY

EXAMPLE:

```
> OPTIONS              : list the currently installed options
>
WIDE BAND JITTER
JITTER SPECTRUM ANALYZER
JITTER GENERATION

>
```

PAGE (pagenumber) : print page # (pagenumber)

This command causes the FIREBERD to print the specified page of the on-line manual to the terminal screen. If no page is specified, the main HELP page is printed. There are currently seven pages available. They are:

1 : main help page
2 : additional help with command entry
3 : front panel control commands
4 : special front panel commands
5 : remote-control-only commands
6 : BERD-BASIC programming
7 : built-in BERD-BASIC programs

EXAMPLE:

```
> PAGE 6          : print out help for BERD-BASIC programming
>
                  .
                  .  : (the help page is printed)
                  .
>
```

Select Plain English or Hex User Message Printouts

PLAIN ENGLISH	: print the plain english status
PLAIN ENGLISH [ON|OFF]	: select character or hex user message prints

This command allows the user to determine if user messages will be printed as either hex bytes or (code-level-dependent) characters in the **CONTROLS** printout. If **PLAIN ENGLISH ON** (can be abbreviated to **PL. ON**) is selected, then all printable characters in the user message will be printed out. All non-printable characters will be printed in hex in the form "(nn)", where the "nn" is the hex character code. Note that the translation of user message bytes from hex to the corresponding characters is code-level-dependent (that is, determined by the number of data bits in the selected character format -- seven for ASCII). Otherwise if **PLAIN ENGLISH OFF** (can be abbreviated to **PL. OFF**) is selected, the **CONTROLS** print will list the user message as hex bytes.

See also: **PRI**

EXAMPLE:

```
> DAT USE                 : select user message data
> PL. ON                  : select plain english printouts
> TIM MOD ASY             : and go into async mode
> CHA FOR DAT 7           : and select ASCII code level
> AUX 41 1/61 62 63       : and set user message 1 to "abc"
> CONTROLS                : and then get a controls print
>

CONTROLS PRINT
   .
   .                      : notice that the user message is listed as "abc"
   .
> PL. OFF                 : then set the user message printouts to hex
> CONTROLS                : and get a new controls print
>

CONTROLS PRINT
   .
   .                      : notice that he user message is now listed in
   .                      : hex as "61 62 63"
>
```

Printer Enable and Function

PRI	: print the state of the printer
PRI [ON|OFF]	: enable or disable printer outputs
PRI [CON|RES]	: generate controls or results printouts
RESULTS	: alternate form for **PRI RES**
CONTROLS	: alternate form for **PRI CON**
CLEAR FIFO	: clear print buffer like **PRI OFF**

These commands manipulate the printer functions. When the **PRI**nter **OFF** selection is made, no printouts will be generated except those specifically requested by the **PRI CON** or **PRI RES** commands. Additionally, when the printer is switched from **ON** to **OFF**, all printouts in process are halted and the printer buffer is cleared. The **PRI RES** and **PRI CON** commands generate results printouts and controls printouts, respectively.

EXAMPLE:

```
> RESULTS                    : generate a results print
>
MANUAL          PRINT
   .
   .                          : (and the results print occurs)
   .

> PRI ON                     : master enable for all automatic printouts
> PRI CON                    : then get a controls print
>
CONTROLS PRINT
   .
   .
   .
>
```

Print Event Select

PRI EVE	: print the print event status
PRI EVE ERR (errortype)	: select print on error
PRI EVE SYN LOS [OFF|ON]	: select print on sync loss
PRI EVE TES INT [OFF|ON]	: select print on test interval
PRI EVE TIM [OFF|(minutes)]	: select print on time interval
PRI EVE DEL	: select Delta prints

This command, along with the **PRI** command, determines when the FIREBERD generates automatic results prints. If **PRI EVE ERR OFF** is specified, no printouts are generated as a result of a received error. If **PRI EVE ERR (errortype)** is specified, a results print is generated for each **(errortype)** detected per error analysis second (E A SEC). The **(errortype)** may be any of the following:

BIT	:bit error
BLK	:block errorf
BPV	:bipolar violation
FRA	:frame error
CRC	:CRC error
CLK	:clock slip
JIT	:jitter hit

The **PRI EVE SYN LOS ON** selection will cause a pre-sync loss results printout to be generated automatically following a sync loss. The **PRI EVE SYN LOS OFF** selection disables this function. Likewise, the **PRI EVE TES INT** selection enables or disables automatic results printout generation at the end of each test interval. The **PRI EVE TIM (minutes)** enables an automatic results print to be generated every **(minutes)** minutes, where **(minutes)** is an integer from 1 to 1440. The **PRI EVE DEL** command enables the Delta printout feature of the FIREBERD. Note that more than one print event may be enabled at a time (with the exception of the **PRI EVE DEL** selection, which disables all other selections automatically).

Note that automatic results printouts will only be generated if the main **PRI**nter enable is set **ON**.
See also: **PRI**

EXAMPLE:

> PRI EVE TIM ON 25	: generate a results print every 25 minutes
> PRI EVE TES INT ON	: and at the end of every test interval
> PRI EVE ERR OFF	: do not generate a printout because of errors
> PRI EVE SYN LOS OFF	: or because of sync losses
>	

Print the Single FIREBERD Result Value

PRINT (result) : print the (result)
PRINT (result)/(string) : print the (result) labeled with the (string)
? ... : alternate short form of **PRINT** ...

This command is used to print out the value of either a BERD-BASIC program variable or a FIREBERD result. If the **PRINT (result)/(string)** option is selected then the result value will be printed with the specified string on its left. The valid **(result)** names are as follows:

%ACT SEC : percentage of active seconds
%AVL SEC : percentage of available seconds
%DEG MIN : percentage of degraded minutes
%EFS : percentage of error-free seconds
%MAS : maximum percentage of jitter mask
%SES : percentage of severely errored seconds
%SYN SEC : percentage of sync seconds
%SVS : percentge of severly violated seconds
%VF EAS : percentage of violation free error analysis seconds
%VFS : percentage of violation free seconds
1SEC CRC : one second CRC errors
1 SEC JTR : maximum jitter unit intervals over previous second
AIS SEC : AIS seconds
AVG BER : average bit error rate
AVG CRC : average CRC error rate
AVG BPVR : average bipolar violation rate
AVG BLER : average block error rate
AVG FER : average framing error rate
AVG FWER : average frame word error rate
AVL SEC : available seconds
BER : bit error rate
BIT ERR : bit errors
BLK ERR : block errors
BLO : blocks
BPV RAT : bipolar violation rate
BPVS : bipolar violations
C-D CHA : clock-data phase changes
CHA ERR : async character errors
CLK LOS : clock losses
CLK SLI : clock slips
CRC E RT : CRC error rate
CRC ERR : CRC errors
DAT LOS : data losses
DAT : date
DEG MIN : degraded minutes
DEL : delay measurement
EA SEC : error analysis seconds

EF EAS	: error free error analysis seconds
EFS	: error-free seconds
ELA SEC	: elapsed seconds
ERR EAS	: errored error analysis seconds
ERR SEC	: errored seconds
FE RAT	: framing error rate
FRA ERR	: framing errors
FRA LOS	: frame sync losses
FW ERR	: frame word errors
FW E RATE	: frame word error rate
GEN FRE	: generator clock frequency
JTR FRE	: jitter frequency list
JTR HIT	: jitter hits
MAX JTR	: maximum jitter unit intervals
PWR LOS	: power losses
RCV COD	: received code type
RCV FRE	: receiver frequency
SES	: severely errored seconds
SIG LOS	: signal losses
SIGL SEC	: signal loss seconds
SIT ID	: site identification
SVS	: severely violated seconds
SYN LOS	: sync losses
SYNL SEC	: sync loss seconds
TIM	: time of day
UNA SEC	: unavailable seconds
VF EAS	: violation-free error analysis seconds
VFS	: violation-free seconds
XSO SEC	: excess zero seconds
YEL SEC	: yellow seconds
(variable)	: BERD-Basic variable

EXAMPLE:

```
> PRINT SIT ID                          : print the FIREBERDs site
                                          identification
>
ALPHA
> ? ELA SEC/"TEST SECONDS"              : print the number of elapsed test seconds
>
TEST SECONDS 43
> PRINT Y                               : print the value of BERD-Basic
                                          variable Y
>
12345
> PRINT A/"LOOPS ="                     : print the value of the variable A
>
LOOPS = 12
>
```

Run Built-in BERD-BASIC Programs

PROG	: run the main interactive menu
PROG AUX	: configure aux functions
PROG FRONT PANEL	: configure the front panel functions
PROG INTERFACE	: configure the interface setup
PROG JITTER	: configure the jitter setup
PROG PRINTER	: configure the printer functions
^ ...	: short from of **PROG** ...

This command is used to access the built-in BERD-BASIC programs. These programs provide access to functions not otherwise available thru simple remote control commands. Running any of these programs does not in any way affect any user BERD-BASIC programs. Page 7 of the on-line mini-manual is reserved for information on built-in BERD-BASIC programs.

These programs are interactive in nature; that is, they generally ask the user questions and print all of the available choices (and, if applicable, the currently selected choice). The user then need only respond with the desired choice followed by a statement terminator (or simply a statement terminator to accept the currently selected choice). The program will continue in this manner until all requirements have been satisfied.

To configure the entire FIREBERD interactively, the following procedure is suggested:

1) Type the **PROG** command at the prompt to enter the interactive menu
2) select **INTERFACE** configuration and answer questions
3) select **AUX** configuration and answer questions
4) select **FRONT PANEL** configuration and answer questions
5) select **JITTER** configuration (if desired) and answer questions
6) select **PRINTER** configuration and answer questions
7) make no selection to exit the main interactive menu

This order is suggested to provide the FIREBERD with the most pertinent set-up information (which interface to use) early in the session.

Since an interactive session can be quite lengthly, an example will not be listed here. Instead, a full annotated listing of the above mentioned procedure will be provided in section 5 of this handbook.

Enable or Disable the Remote Control Command Prompt

PROMPT	: print the status of the prompt
PROMPT [ON|OFF]	: turn the prompt on or off

This command allows the user to selectively enable or disable the remote control command prompt. If the prompt is turned **ON**, a ">" character will be sent to the controller whenever the FIREBERD is ready to receive another command. If the prompt is turned **OFF**, no such transaction occurs. Note that the prompt character will change to a "+" or "<" whenever the FIREBERD is in either a **HOLD** or **ENTER** mode respectively. This warns the user that the FIREBERD is in a non-standard mode.

See also: **HOLD, ENTER**

EXAMPLE:

> PROMPT OFF	: turn off the prompt
BEEP	: then make the FIREBERD beep
PROMPT ON	: then turn the prompt back on
> HOLD	: and enter printer HOLD mode
+ REL	: then release the printer HOLD

>

Recall Front Panel Program

REC (prognum) : recall the desired front panel program

This command recalls the front panel program number **(prognum)** from NOVRAM, where **(prognum)** is between 0 and 9. If that program specifies an interface that is not currently installed into the FIREBERD, then a warning message will be generated. The warning message will repeated following every entered command until the situation is remedied (by either installing the interface indicated by the **CONTROLS** printout or by selecting a different interface with the **INT SET** or **REC** command).

See also: **STO, INT SET**

EXAMPLE:

```
> REC 1                    : recall front panel program 1
> REC 3                    : recall front panel program 3
Warning: different interface installed.

>
```

REL

This command ends a printer hold initiated by a **HOLD** command, allowing all accumulated results to be printed. Note that the **LOCAL** and **TERMINAL** commands perform an automatic printer **REL**ease as part of their function. This ensures that printouts cannot be inadvertantly held when the FIREBERD changes modes.

See also: **HOLD**

EXAMPLE:

```
> HOLD                          : hold all printer outputs for now
> CHA FOR                       : request the status of the CHAr FORmat switch
> REM NO PRINTOUTS ARE GENERATED YET
> REL                           : then release the printer hold
>
DAT 7
PAR EVE                         : and out comes the printout requested earlier
STO 2

>
```

REM (remark or comment)

This command, and any remarks or comments following it on the same line, is ignored by the FIREBERD. It can be used for documenting programs or as a simple reminder for anything.

EXAMPLE:

```
> REM THIS IS A COMMENT!
> REM SO IS THIS
> REMEMBER ME?                    : all REM statements are ignored
>
```

Restart Test

RES

This command restarts the FIREBERD test.

EXAMPLE:

```
> RES                    : restart the test in progress
>
```

RETURN

This command indicates the end of a subroutine. When the FIREBERD encounters a **RETURN** statement in a running program, it goes back to executing the commands immediately following the most recently encountered **GOSUB**. The **GOSUB** and **RETURN** statements are always executed in pairs. That is, a **GOSUB** command causes the FIREBERD to execute subsequent commands starting at the first line of the specified subroutine. The **RETURN** command, on the other hand, tells the FIREBERD that the subroutine is finished and command execution should continue from where it left off following the **GOSUB** command.

See also: **GOSUB**

EXAMPLE:

```
> 10 LPRINT "CALLING LINE 90"     : make a program to
> 20 GOSUB 90                     : execute the subroutine starting at
                                    line 90
> 30 LPRINT "I'M BACK"
> 40 END                          : then end
> 90 REM THIS IS IT!              : the subroutine prints its message and
> 100 LPRINT "I'M IN THE SUBROUTINE!"
> 110 RETURN                      : then returns back to the main program
> RUN
CALLING LINE 90
I'M IN THE SUBROUTINE
I'M BACK

* DONE *
>
```

Manipulate the RLSD/RR Control Line

RLS	: print the status of the RLSD/RR control line
RLS [ON|OFF]	: set the new state of the RLSD/RR control line
RR ...	: alternate form of RLS ...

This command displays or sets the state of the RLSD (for RS-232) or RR (for RS-449) interface control line. It is only valid when FIREBERD is operating in the DCE mode.

EXAMPLE:

```
> RLS                    : print the current state of the RLSD line
>
OFF

> RLS ON                 : then turn it on
>
```

Manipulate the RTS/RS Control Line

RTS : print the status of the RTS/RS control line
RTS [ON|OFF] : set the new state of the RTS/RS control line

This command displays or sets the state of the RTS (for RS-232) or RS (for RS-449) interface control line. It is only valid when the FIREBERD is operating in DTE mode.

EXAMPLE:

```
> RTS                    : print the current state of the RTS line
>
OFF
> RTS ON                 : then turn it on
>
```

Run the Current Program

RUN : run from start of program
RUN (line number) : run, starting from the specified line

This command causes the FIREBERD to begin executing the working program. If the **(line number)** is omitted, the execution will start at the first line of the program (that is, the one with the lowest line number). Otherwise, execution will start at the specified **(line number)**. Program execution will normally proceed from the lowest numbered lines to the highest numbered lines, in sequence. This can be altered by the user through the use of the **GOTO, GOSUB,** and **IF-THEN** commands.

EXAMPLE:

```
> 10 LPRINT "THIS IS A TEST"      : write a simple program with two lines
> 20 LPRINT "SO IS THIS"
> RUN                             : and run it
THIS IS A TEST                    : the statements are executed
SO IS THIS

* Done *                          : and the program finishes
>
```

SAVE (program name)

This command causes a copy of the working program to be saved to ramdisk under the specified **(program name)**. This copy may later be retrieved, unchanged, using the **LOAD** command. Note that the ramdisk is only large enough to accomodate four saved programs at once. The **KILL** command must be used to removed unwanted programs, thus making space for new ones.

See also: **DIR, KILL, LOAD**

EXAMPLE:

```
> 10 REM THIS PROG
  IS SAVED          : make a program to save in permanent storage
> SAVE DOG          : and save it under the name DOG
> NEW               : then wipe out the working program
> 10 REM NEW
  GARBAGE           : and enter a meaningless new one
> LOAD DOG          : then load the original program back
> LIST              : and check to see if it is there
>

10 REM THIS PROG IS SAVED

[xxx bytes free]
>
```

Self-Loop Select

SEL LOO : print the self-loop status
SEL LOO [ON|OFF] : select the self-loop state

This command is used to print and manipulate the self-loop state. The command **SEL LOO ON** puts the FIREBERD into Self-Loop mode. Correspondingly, **SEL LOO OFF** takes the FIREBERD out of Self-Loop mode.

EXAMPLE:

```
> SEL LOO                 : print the self-loop status
>
OFF

> SEL LOO ON              : put the FIREBERD into self-loop
>
```

Print Automatic Self-Test Status

This command prints the status of the automatic self-test. The status messages that can be sent in response to the **SELF TEST** command are:

NO SELF TEST ERRORS
NOVRAM SELF TEST ERROR
EPROM SELF TEST ERROR
RAM SELF TEST ERROR

EXAMPLE:

>**SELF TEST** :check the status of the automatic self-test

NO SELF TEST ERRORS
>

Store Front Panel Program

STO(prognum)

This command stores the current front panel configuration under front panel program number **(prognum)**, where **(prognum)** is between 0 and 9, inclusive.

See also: **REC**

EXAMPLE:

```
> STO 7                    : store the current front panel configuration in prog 7
>
```

Stop Program Execution

STOP

When this command is encountered in a running program, it causes the FIREBERD to halt the program's execution. The user may then enter commands manually and, if desired, resume the program's execution from where it left off (following the **STOP**) by using the **CONT**inue command. Note that a program may not be be **CONT**inued if any of its lines have been changed.

See also: **CONT**

EXAMPLE:

```
> 10LPRINT "FIRST HALF"          : that runs in two halves
> 20 STOP                        : with a STOP in the middle
> 30 LPRINT "SECOND HALF"
> 40 END
> RUN                            : then run the program
FIRST HALF                       : and the first half finishes

Break in Line 20

> CONT                           : then continue from after the STOP
SECOND HALF                      : and the program finishes

*Done*
>
```

Set Synthesizer Frequency

SYN FRE : print the current synthesizer frequency
SYN FRE (frequency) : set the new synthesizer frequency

This command sets and displays the synthesizer frequency in kHz. The **(frequency)** field can be any valid frequency between .05 kHz and 15000 kHz.

EXAMPLE:

> **SYN FRE** : display the current synthesizer frequency
>
9.6 (kHz)

> **SYN FRE 1544** : then set the synthesizer to 1.544 MHz
>

Sync Lost Indicator Reset

SYN LOS

This command simply resets the SYNC LOST indicator on the FIREBERD front panel.

EXAMPLE:

```
> SYN LOS                    : make sure the indicator is off
>
```

Timing Mode Select

TIM MOD : print the current timing mode
TIM MOD [SYN|ASY|REC] : select the new timing mode

This command sets the FIREBERD timing mode to **SYN**chronous, **ASY**nchronous, or **REC**overed. Note that certain external interfaces can only operate in one mode. In such a case, the FIREBERD will select that mode automatically without any user intervention.

EXAMPLE:

```
> TIM MOD                    : print the current timing mode
>
ASY

> TIM MOD SYN                : then select synchronous timing mode
>
```

Set State of Rear Panel Test Point Z

TPZ : print the state of the test point
TPZ [HIGH|LOW] : set the state of the test point

This command is used to manipulte the state of the rear panel test point. Selecting **TPZ HIGH** will raise the test point to +5 volts; likewise, selecting **TPZ LOW** will lower the test point to 0 volts. This test point may be changed by a BERD-BASIC program in response to various test conditions thus providing external indications of alarm conditions or other statuses.

EXAMPLE:

```
> 10 INT SET DS1/FE          : we are using the DS1/T1/Fe interface
> 20 IF FRM SYNC THEN 50     : check if we have frame sync
> 30 TPZ LOW                 : force TPZ low if not
> 40 GOTO 20
> 50 TPZ HIGH                : else force TPZ high
> 60 GOTO 20                 : then go back and wait more
> RUN                        : then let TPZ monitor frame sync

/
Break in line 20             : (press Control C to stop the program)

>
```

Load User Message With ASCII Text

USER [1|2|3]

This command allows the user to enter ASCII text from the terminal directly into user message **1**, **2**, or **3**. All characters received by the FIREBERD following a **USER** command will be stored as the selected user message until an ASCII ETB (Control W, character code 17H) is encountered, at which point the FIREBERD returns back to normal command mode. Note that the ETB is not, and cannot be, stored as part of the user message using the **USER** command.

See also: **AUX 41**

EXAMPLE:

```
> USER 2                    : tell the FIREBERD to load user message 2
Enter text followed by a Control W.
> THIS IS IT!               : then type "THIS IS IT!" followed by a
                              carriage return, linefeed, and ETB
                              (Control W)
> AUX 41 2                  : then find out what is in user message 2
>
54 48 49 53 20    49 53 20 49 54    21 0D 0A

>
```

Wait for Sync-Up or for a Specified Time

WAIT SYNC	: wait for the FIREBERD to sync up
WAIT (number)	: wait from 0.1 to 65534.9 seconds

This command causes the FIREBERD to wait for the specified time before resuming a program. If the **WAIT SYNC** option is specified, then the FIREBERD will not resume the program until synchronization is achieved. This is useful in BERD-BASIC programming to ensure that results are actually available before printing or testing them (since an unavailable result may produce an error and abort program execution). Note that a Control C received from the controller will break out of a **WAIT** statement.

EXAMPLE:

```
> 10 PRINT TIM          : print the time of day
> 20 WAIT 60            : then wait a minute
> 30 PRINT TIM          : then print the time again
> RUN

   11:00:30             : it is 11:00 AM
   11:01:30             : now it's one minute later

*Done*
>
```

SECTION 3
FIREBERD ERROR MESSAGES

An **ERROR** message indicates that the FIREBERD found something wrong with a remote control command and, therefore, did not process it.

ERROR: An empty slot cannot be selected.
The **ISU SELECT** command was used to select an empty slot in the ISU 6000.

ERROR: Can't continue.
The user is attempting to **CONT**inue either a built-in BERD-BASIC program or a user program that has been altered (that is, lines have been changed) since the last **STOP** command or control C. This is not allowed.

ERROR: Characters after statement end.
The FIREBERD found additional characters following what it thought was the end of a legal command.

ERROR: Command buffer overflow.
Either the user sent a command that was greater than 512 characters in length, or the user sent too many commands too fast.

ERROR: Command not accepted, switch contention.
A contention exists involving the specified "front panel" switches. One or more of them must be changed (or a front panel program recalled) to remedy the situation before proceeding with other functions.

ERROR: Command not allowed in a program.
The user is attempting to execute a command in a BERD-BASIC program which can only be executed as a direct command from the remote controller.

ERROR: Command not valid in this mode.
The user attempted to perform a function that is not applicable in one of the modes that the FIREBERD is currently operating in.

ERROR: Command only allowed in a program.
The user is attempting to execute a command directly from remote control which is only allowed when used as a command in a BERD-BASIC program.

ERROR: File not found.
A ramdisk program name has been specified that does not belong to one of the programs that is currently saved.

ERROR: GOSUBs nested more than 10 deep.
Most likely, a subroutine is accidentally calling itself or a **RETURN** statement has been left out. Subroutines are not ending.

ERROR: IF not followed by THEN.
The syntax of the **IF-THEN** statement has been violated. The **THEN** part was not found.

ERROR: Illegal delimiter.
A delimiter ("/", "=", etc.) was expected in the command, but none was found.

ERROR: Illegal expression.
The final result of an expression being assigned to a variable must be within the range of 0 to 32767. Intermediate results during expression evaluation may, on the other hand, range from -32768 to 32767.

ERROR: Illegal file name.
A ramdisk file name must be a **string** from 1 to 8 characters in length.

ERROR: Illegal line number.
Either a non-existent BERD-BASIC program line was referenced (by a **GOTO, GOSUB**, etc.) or the user attempted to enter a program line with a number greater than 65534.

ERROR: Illegal relationship.
The **IF-THEN** command has a **relationship** specified that could not be recognized.

ERROR: Illegal result/variable name.
The result or variable specified was illegal. A list of valid result names can be obtained using the **HELP PRINT** command; a variable name must be a single letter from "A" to "Z".

ERROR: Illegal value.
The value supplied with the command is not within the legal range of the requested function.

ERROR: Inconsistent with current configuration.
Either an option that is not present is required to perform the requested function, or the FIREBERD is in a mode where the requested function is not applicable.

ERROR: Inconsistent with interface or interface mode.
The requested command is not supported for the interface currently selected by the **INT SET** command or the interface is in such a mode that the command is not applicable.

ERROR: Inconsistent with selected timing mode.
The user is attempting to perform a function which does not pertain to the current timing mode selected with the **TIM MOD** command.

ERROR: Inconsistent with this interface.
The user is attempting to perform a function which is not possible using the data interface currently selected by the **INT SET** command.

ERROR: Inconsistent with this option.
The requested function does not apply to the option installed.

ERROR: ISU is installed. No external interface solution is allowed.
The **INT SET EXT** command was used while an ISU-6000 was installed.

ERROR: ISU is not selected.
The status of the **ISU** command does not apply because the ISU is not selected.

ERROR: Macros nested more than 5 deep.
Most likely, a macro is attempting to recall itself (which, in turn, attempts to recall itself again, ad infinitum). This is not allowed.

ERROR: Nothing saved.
The user is attempting to save an empty program onto the ramdisk.

ERROR: Option not installed.
The requested function requires an option that is not installed in the mainframe. For a list of options, see the **OPTION** command.

ERROR: Program memory overflow.
The user attempted to enter a program that was too long to fit into memory. Either the program must be shortened (that is, taking blank characters out of lines, using command abbreviations, etc.) or the program must be broken into separate pieces that are **CHAIN**ed together.

ERROR: Receiver framing error.
The FIREBERD detected one or more async framing errors in the input command. The controller's data format may be different from that of the FIREBERD.

ERROR: Receiver overrun error.
The controller is sending characters too fast for the FIREBERD or is sending characters while the **clear to send** control signal is being output false (indicating that the FIREBERD is not ready for more data). Solutions include responding to signaling and/or lowering the baud or data transfer rate.

ERROR: Requested data interface not installed.
The interface specified by the **INT SET** command is not currently installed in the FIREBERD.

ERROR: Requested result not legal here.
The user is attempting to print, display, or compare (in an **IF-THEN** statement) a result for which the function is not supported.

ERROR: Result not applicable.
The requested result is not applicable in the current timing or interface mode.

ERROR: RETURN without GOSUB.
A **RETURN** statement was encountered that was not part of a subroutine; that is, it did not follow a **GOSUB** command which could be **RETURN**ed to.

ERROR: Switch contention.
The command was accepted; however, it caused a contention involving the specified front panel "switches". One of them must be changed (or a front panel program recalled) before proceeding with other functions.

ERROR: Too many files saved.
The user is attempting to save more than four programs on the ramdisk at once. The only remedy is to **KILL** one or more of the programs already on the ramdisk before attempting to save another one.

ERROR: Unrecognized command.
The FIREBERD did not recognize the command name entered in the command line.

ERROR: Unrecognized parameter.
The FIREBERD recognized the requested command but could not interpret the specified command parameter.

A **WARNING** message is to warn the user that the FIREBERD had to do something slightly out the ordinary in order to process the command, but the command was processed.

WARNING: Different interface installed.
The interface specified by the current front panel program is not installed. Either install the interface indicated by the **CONTROLS** print or select a new interface using the **INT SET** command.

WARNING: Extra input ignored.
The user responded to an **INPUT** command with a **string** greater than 31 characters in length. The string was truncated to 31 characters when stored in the user macro specified by the **INPUT** command.

WARNING: Illegal input.
This warning is generated by the **BREAK WARN** command when executed following a BERD-BASIC program error occurring in **BREAK OFF** mode. It generally specifies that the user entered an illegal string into an **INPUT** command.

WARNING: Jitter generation out of range.
The specified jitter amplitude can not be generated at the specified frequency. To find the amplitude of the actual jitter being generated, use the **JIT GEN AMP** command.

WARNING: Receiver parity error.
The FIREBERD received one or more characters in the command with a parity error.

WARNING: Result is not available yet.
The user has requested the user of a result whose value cannot yet be computed. It may be because the FIREBERD is out of sync and a result was specified that is only available when the FIREBERD is in sync. If this occurs in a program, the **WAIT SYNC** command should be issued prior to requesting the result.

WARNING: Selection rounded to allowable value.
The value entered with the command had to be rounded to the nearest allowable value before it could be processed.

WARNING: Synthesizer defaults to 1544 kHz with jitter generation.
When jitter generation modulation is on, the synthesizer frequency is set to 1544 kHz and cannot be changed.

WARNING: To really clear NOVRAM, type "AUX 43!".
This forces the user to type an additional command before the FIREBERD will really clear all of its NOVRAM.

WARNING: To really clear AUX functions, type "AUX 99!".
This forces the user to type an additional command before the FIREBERD will really clear AUX functions 1 thru 11.

The FIREBERD will generate a **NOTICE** message to inform the user of certain events that might otherwise go unnoticed.

NOTICE: Single analysis mode test complete, results frozen.
Single analysis mode has been selected (using the **ANA MOD SIN** command) and the test interval is finished. All results (including **TIME, DATE** and **LEDS**) have been frozen to preserve their state. The results will not change again until a major switch is changed (including changing to continuous analysis mode using the **ANA MOD CON** command).

SECTION 4
FIREBERD 6000 ON-LINE HELP DIRECTORY

All commands to the FIREBERD consist of an ASCII command name followed by optional ASCII parameter selections and either a carriage-return or linefeed. Spaces (" ") and letter case (lower/UPPER) are ignored.

HELP!	Print a list of all available command names.
HELP (command name)	Print the PARAMETER SYNTAX for a command.
HELP AUX	Print a list of all available AUX functions.
PAGE 1 or HELP or ?	Print this page.
PAGE 2	Print additional HELP for command entry.
PAGE 3	Print HELP for front panel switch commands.
PAGE 4	Print HELP for special front panel commands.
PAGE 5	Print HELP for "remote control only" commands.
PAGE 6	Print HELP for BERD-BASIC programming.
PAGE 7	Print HELP for built-in BERD-BASIC programs.
(command name) (parameter)	Set the new state of a command.
(command name)	Print the current state of a command.

The following character have special meaning to the FIREBERD:

<BREAK>	(Three times slowly) Enter auto-baud mode. Type spaces (" ") repeatedly (~ 5 Hz) until auto-baud achieved, then <ESCAPE>.
<control h>	BACK SPACE – deletes the previous character.
<control x>	CANCEL – cancels the current input line.
<control c>	Cancels the input line, clears the print fifo, aborts any loop codes, and breaks any program execution.
<control s>	XOFF – suspends all printer outputs until XON.
<control q>	XON – resumes all printer outputs from XOFF

"*"	Recalls the help/status macro.
"&n"	Recalls user macro #n (where n = 0 to 9).
":"	Separates multiple commands on the same line.
"," or " – "	Separates multiple parameter selections of a command.
"#nn/"	Selects command/parameter #nn as listed by the NUMBER command. ("/" is optional if the character that follows is not numeric).
"{. . .}"	Surround comments to be ignored by the FIREBERD.
"-"	Repeats the previous command.
"."	Follows an abbreviated command or parameter name.

Front panel switch manipulation commands include:

ANA MOD	– select analysis mode.
ANA RES 1 and 2	– select left and right display results, respectively.
AUX (nn)	– select auxilliary function number nn.
CHA FOR	– select async character format.
DAT	– select generator data pattern.
ERR INS	– select error insertion rate.
GEN CLK	– select generator clock source.
INT SET (int)	– select data interface setup.
JIT	– select jitter setup.
PRI EVE	– select print event.
PRI	– select printer power or results/controls.
REC/STO	– recall/store front panel program.
RES	– restart test.
RTS/DTR	– set interface signals (also RLS/DSR, RR/DM, or LOOUP/LOODN).
SEL LOO	– select self loop mode.
SYN FRE	– select synthesizer frequency.
TES INT	– select the test interval length.
TIM MOD	– select the timing mode.

Special front panel switch commands perform the same function as the normal front panel switch commands but with a simplified comand entry format. They include:

CLEAR FIFO	– clear the printer fifo.
CONTROLS	– generate a controls print.
CUSTOM	– select the custom printout status.
ERROR	– insert a single error in the TX data.
FAST/SLOW	– select fast/slow RS-232 printer mode.
FORMAT/UNFORMAT	– select formatted/unformatted printer mode.
HELLO	– print the software revision level.
LONG	– select long results print mode.
OCT TIM	– select octet timing mode for 64k G.703 interface.
RESULTS	– generate a results print.
STANDARD	– select the standard results print.
STATUS	– select the customized status messages.
TERM 232	– select the RS-232 printer line terminator.
TERM 488	– select the IEEE-488 printer line terminator.
TIME/DATE	– set or display the time/date.
WIDTH	– select the printer width.

The remote-control-only special functions include:

AUDIO	– select the audio level override.
BEEP	– make both the terminal and FIREBERD beep.
CLS	– clear the terminal screen.
DEVICE CLEAR	– initialize the FIREBERD to its power-up state (also BYE).
DISPLAY	– select the front panel display switch mode.
ECHO/PROMPT	– select the echo/prompt function status.
ENTER/END	– disable/re-enable contention error messages.
HOLD/REL	– hold/release all printer outputs
ISU	– control the Interface Switching Unit.
LEDS	– print state of front panel LEDs.
LOCAL or /	– return the FIREBED to local mode.
LPRINT	– print a literal string.
MACRO	– define or display user macros.
PLAIN ENGLISH	– select plain english or hex user message prints.
PRINT or ?	– print the result or variable value.
SELF TEST	– print automatic self-test status.
TERMINAL or .	– set the FIREBERD up to talk to a dumb terminal.
TPZ	– control rear panel test point "Z".
USER	– enter a user message in ASCII.

The FIREBERD accepts the following BERD-BASIC commands:

nnnn (command)	– load remote control command as line number nnnn.
nnnn	– delete program line number nnnn.
BREAK	– select program action on error.
CHAIN	– chain execution to a program in permanent storage.
COMMON	– preserve all variables thru the next CHAIN.
DIR	– print the names of all files in permanent storage.
GOSUB/RETURN	– Goto and reutrn from subroutine, respectively.
GOTO	– transfer program execution to the specified line.
IF (... THEN)	- GOTO specified line if condition is true.
INPUT	– enter line from terminal into user macro.
KILL	– delete a program from premanent storage.
LET	– (optional) assign a new value to a variable.
LIST	– list the current program.
LOAD/SAVE	- load/save a program from/to permanent storage.
NEW	– clear the current program in preparation for a new one.
RUN/CONT	– run/continue the current program.
STOP/END	– stop/end the program execution.
WAIT	– wait for SYNC or the specified number of seconds.

The built-in BERD-BASIC programs provide additional capabilities not otherwise available through simple remote control commands. Note that these programs may modify the user macros. They include:

PROG OR ^	– run the main interactive menu.
PROG AUX	– configure AUX functions.
PROG FRONT PANEL	– configure the front panel functions.
PROG INTERFACE	– configure the interface setup.
PROG JITTER	– configure the jitter setup.
PROG PRINTER	– configure the printer functions.

SECTION 5
SAMPLE INTERACTIVE CONFIGURATION SESSION

```
>
> PROG

---------- Main Program Menu ----------

Built-in programs

    AUX         FRONT PANEL         INTERFACE         JITTER
    PRINTER

Run which one (<CR> to exit)? INTERFACE

Use internal RS-232 interface (Y/N)? N

Using DS1/FE interface

Select result-mode
    STD
    LIV
Current selection is
    STD
New selection (<CR> to pass)?          (user presses carriage return)

Select thru-mode
    ON
    OFF
Current selection is
    OFF
New selection (<CR> to pass)?          (user presses carriage return)

Select input-impedance
    TER
    BRI
    DSX MON
Current selection is
    TER
New selection (<CR> to pass)? BRI

Select line-code
    AMI
    B8ZS
Current selection is
    AMI
New selection (<CR> to pass)? B8ZS
```

Select framing-format
OFF
D4
FE
Current selection is
OFF
New selection (<CR> to pass)? **D4**

Select line-build-out
0
-7.5
-15
Current selection is
Current selection is
0
New selection (<CR> to pass)? **(user presses carriage return)**

Select BPV-insertion
OFF
SIN
RAT xE-y
where: x = mantissa (1 to 9)
y = exponent (3 to 9)
Current selection is
OFF
New selection (<CR> to pass)? **RAT 1E-6**

----------Main Program Menu----------

Built-in programs

AUX FRONT PANEL INTERFACE JITTER
PRINTER

Run which one (<CR> to exit)? **FRONT PANEL**

Select generator-clock-source
SYN
INT
BNC
Current selection is
SYN
New selection (<CR> to pass)? **(user presses carriage return)**

Select synthesizer-frequency
ffffff
where: f = .05 to 15000 kHz
Current selection is
9.6 (kHz)
New selection (<CR> to pass)? **1544**

```
Select transmit-data
      MAR
      1:1
      63
      511
      2047
      2^15-1
      2^20-1
      2^23-1
      QRS
      PRG
      FOX
      USE
Current selection is
      63
New selection (<CR> to pass)? QRS


Select test-interval
      10^4
      10^5
      10^6
      10^7
      10^8
      10^9
      10^10
      TIM   hh:mm:ss
Current selection is
      10^5
New selection (<CR> to pass)? 10^6


Select analysis-mode
      SIN
      CON
Current selection is
      CON
New selection (<CR> to pass)?          (user presses carriage return)

-----------------------------------Main Program Menu-----------------------------------


Built-in programs

      AUX         FRONT PANEL         INTERFACE         JITTER
      PRINTER

Run which one (<CR> to exit)? AUX
```

Available AUX functions

01 (gen clk pol)	02 (rcv clk pol)	03 (act syn loss)
04 (single trans)	05 (intf control)	06 (gen data inv)
07 (rcvr clk sel)	08 (out band flo)	09 (in band flow)
10 (syn loss thr)	11 (printer sel)	30 (block length)
31 (delay meas)	32 (common freqs)	33 (pattern)
34 (site id)	35 (result sel)	36 (status sel)
37 (print format)	38 (RS-232 rem)	39 (IEEE-488 rem)
40 (software ver)	41 (user mes sel)	42 (pre syn loss)
43 (clear novram)	44 (histogram)	60 (time/date)
99 (clear in use)		

Select AUX function (<CR> to exit)? **3**

CLE
HAL (*)
CON (*)
AUT DIS (*)

Current selection is

AUT DIS (*)

New selection (<CR> to pass)? **CON**

Available AUX functions

01 (gen clk pol)	02 (rcv clk pol)	03 (act syn loss)
04 (single trans)	05 (intf control)	06 (gen data inv)
07 (rcvr clk sel)	08 (out band flo)	09 (in band flow)
10 (syn loss thr)	11 (printer sel)	30 (block length)
31 (delay meas)	32 (common freqs)	33 (pattern)
34 (site id)	35 (result sel)	36 (status sel)
37 (print format)	38 (RS-232 rem)	39 (IEEE-488 rem)
40 (software ver)	41 (user mes sel)	42 (pre syn loss)
43 (clear novram)	44 (histogram)	60 (time/date)
99 (clear in use)		

Select AUX function (<CR> to exit)? **(user presses carriage return)**

----------Main Program Menu----------

Built-in programs

AUX FRONT PANEL INTERFACE JITTER
PRINTER

Run which one (<CR> to exit)? **JITTER**

Select wide-band-jitter

FUL
>8KHZ
n.n where: n = 0.1 to 6.5 UI jitter hit threshold

Current selection is
    FUL
    6.5 (UI)
New filter-bandwidth (<CR> to pass)? **>8KHZ**
New hit-level (<CR> to pass)? **4.2**

----------Main Program Menu----------

Built-in programs

    AUX FRONT PANEL INTERFACE JITTER
    PRINTER

Run which one (<CR> to exit)? **PRINTER**

Enable automatic printouts (Y/N)? **Y**
Delta or normal results prints (D/N)? **N**

Select print-on-sync-loss
    OFF
    ON
Current selection is
    ON
New selection (<CR> to pass)? **OFF**

Select print-on-test-interval
    OFF
    ON
Current selection is
    OFF
New selection (<CR> to pass)? **(user presses carriage return)**

Select print-on-error
    OFF
    SEC
    BLK
    BPV
Current selection is
    SEC
New selection (<CR> to pass)? **OFF**

Select print-on-time-interval
    OFF
    nnnn where: n 1 to 1440 mins
Current selection is
    OFF
New selection (<CR> to pass)? **10**

Custom, standard, or long results prints (C/S/L)? **S**

Enable any status printouts (Y/N)? **Y**
Select status-messages to enable

ALL	CLK LOS	DAT LOS	FRA ERR
NEW CON	OVE PRI	PHA CHA	POW UP
SIG CHA	SIG LOS	SYN ACQ	SYN LOS
TES RES			

Selection (<CR> to pass)? **TES RES**
Selection (<CR> to pass)? **OVE PRI**
Selection (<CR> to pass)?

----------Main Program Menu----------

Built-programs

AUX	FRONT PANEL	INTERFACE	JITTER
PRINTER			

Run which one (<CR> to exit)? **(user presses carriage return)**

>
>

SECTION 6
SAMPLE IEEE-488 PROGRAM

This appendix provides a computer program written for use on a Hewlett Packard 85 computer.

SAMPLE PROGRAM #1: IEEE-488/GPIB-PC Controller Program

This program is for use with the National Instruments GPIB-PCII and BASIC language interface for the IBM PC. Lines 1-99 must come from the DECL.BAS file included in the National Instruments package.

```
100 REM **** GPIB-PC / BASIC PROGRAM TO CONTROL FB6000  ****
110 REM
120 REM LINES 1—99 MUST COME FROM DECL.BAS SUPPLIED WITH
YOUR
130 REM GPIB-PC SOFTWARE PACKAGE
140 REM
150 REM READ GPIB-PC DOCUMENTATION FOR FURTHER INFORMA-
TION
160 REM
170 REM THIS PROGRAM ASSUMES FB6000 IS ADDRESS 1 ON IEEE-488
BUS
180 REM
190 REM ******************* INITIALIZATION ******************
200 DIM C$[255]
210 CLS:KEY OFF:LOCATE 25,1
220 PRINT " PRESS ANY KEY TO ENTER COMMAND MODE —-Q COM-
MAND QUITS"
230 LOCATE 1,1
240 DEV$ = "DEV1" 'ASSUMES DEVICE ADDRESS 1
250 CALL IBFIND(DEV$, DEV%)
260 IF DEV% <0 THEN PRINT DEV$;" not found in GPIB-PC configura-
tion";GOTO 460
270 GOTO 380
280 REM
290 REM ******* ENTER USER COMMAND **********
300 PRINT DEV$;">";MSG$;
310 LINE INPUT MSG2$
```

```
320 MSG$ = MSG$ + MSG2$
330 IF MSG$ = "q" OR MSG$ = "Q" THEN 460
340 MSG$ = MSG$ + CHR$(13) + CHR$(10) 'APPEND <CR> <LF> TO
TERMINATE CMD
350 CALL IBWRT (DEV%,MSG$) 'WRITE ASCII STRING TO FB6000
360 RETURN
370 REM
380 REM ****** POLL CONTINUOUSLY ********
390 MSG$ = INKEY$ "IF LEN(MSG$) <>0 then GOSUB 290 'CHECK
KEYBOARD
400 CALL IBRSP(DEV%, STA%) 'SERIAL POLL THE FIREBERD
410 IF (IBSTA% AND &H4000) THEN 450
420 IF (STA% AND !) <>0 THEN PRINT "SYNTAX ERROR"
430 IF STA% >0 THEN C$=SPACE$(255):CALL IBRD(DEV%,C$):PRINT
LEFT$(C$,IBCNT%);
440 GOTO 380
450 PRINT DEV$;" TIMEOUT—ERR = ";IBERR%;
460 KEY ON:END
```

SAMPLE PROGRAMS 2-3: IEEE-488/HP-85 Controller Programs

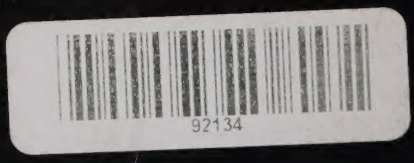
These programs are for use with the Hewlett-Packard HP-85 computer equipped with an HP-IB interface and IO-ROM.

```
10 ! PROGRAM NAME "SRQ488"
20 ! DOES INTERRUPT DRIVEN I/O
30 ! ON IEEE-488 BUS
40 !
50 DIM A$[99],B$[99]
60 ON TIMEOUT 7 GOTO 450
70 SET TIMEOUT 7;1200
80 CLEAR
90 DISP "SRQ DRIVEN IEEE-488 CONTROLLER
100 DISP " PROGRAM"
110 DISP
120 DISP "BE SURE THAT AUX 39, SRQ MODE"
130 DISP "IS SET TO DAT"
140 DISP
150 DISP "PRESS k1 TO ISSUE A COMMAND
160 DISP "PRESS k2 to CHANGE ADDRESS"
170 DISP
180 GOSUB 510
190 ! GET ANY CHR FROM BEFORE
200 ! SRQ ENABLED
210 ENTER 700+I USING "#%,%K"; A$
220 DISP A$
230 ! ENABLE INTERRUPTS
240 ON INTR 7 GOSUB 390
250 ENABLE INTR 7;8
260 ON KEY# 1 GOSUB 300
270 ON KEY# 2 GOSUB 510
280 GOTO 280
290 !
300 ! ** OUTPUT COMMAND **
310 DISP
320 DISP "COMMAND"
330 INPUT B$
340 OUTPUT 700$+I ;B$
350 OUTPUT 700+I ;"/"
360 RETURN
370 !
380 ! INTERRUPT SERVICE ROUTINE
390 S=SPOLL(700+I)
400 ENTER 700+I USING "#%,%K"; A$
410 DISP A$
```

```
420 ! CLEAR HP-85 INTERRUPTl
430 STATUS 7,1 ; S
440 ENABLE INTR 7;8 ^ RETURN
450 ! ** HANDLE BUS TIMEOUT **
460 ABORTIO 7
470 DISP "BUS TIMEOUT—RESET"
480 BEEP
490 WAIT 1500
500 GOTO 230
510 ! GET NEW DEVICE ADDRESS
520 DISP "ENTER ADDRESS OF DEVICE"
530 INPUT I
540 RETURN
```

```
10 ! PROGRAM NAME "EZ488"
20 ! DESIGNED TO PERFORM I/O
30 ! VIA IEEE-488 BUS
40 !
50 !
60 ! ** INITIALIZATION
70 CLEAR
80 DIM I$[99],0$[99]
90 DISP "MENU"
100 DISP "————————"
110 DISP "1) DEPRESS k1"
120 DISP " TO SEND A COMMAND"
130 DISP "2) DEPRESS k2"
140 DISP " TO SELECT ADDRESS"
160 ON KEY# 2 GOTO 70
170 DISP "————————"
190 DISP "OF YOUR INSTRUMENT"
200 INPUT A
210 IF A <32 THEN 260
220 GOTO 70
230 !
240 !
250 ! ** WAIT FOR STATUS <>0
260 CLEAR
270 GOSUB 590
280 ON TIMEOUT 7 GOSUB 620
290 SET TIMEOUT 7;500
300 !
310 !
330 ! ITS STATUS
340 S = SPOLL (700 + A)
350 !
```

```
360 ! IS DATA AVAILABLE (dav)
370 IF BINAND(S,128) <>0 THEN GOSUB 510
380 GOTO 280
390 !
400 !
410 ! *** SUBROUTINES ***
420 !
430 ! ** SEND COMMANDS TO BUS
440 DISP "ENTER COMMAND"
450 INPUT 0$
460 OUTPUT 700+A ;0$
470 GOSUB 590
480 RETURN
490 !
500 !
510 ! READ DATA UNTIL EOI
520 ENTER 700+A USING "#%,%K"; I$
530 DISP I$
540 RETURN
550 !
560 !
570 DISP "TIME OUT ON BUS"
580 DISP "+++++++++++++++++++++"
590 DISP "WAITING FOR A PRINT"
600 DISP "FROM ADDRESS";A
610 DISP
620 ABORTIO 7
630 RETURN
```



Telecommunications
Techniques
Corporation

444 NORTH FREDERICK AVENUE
GAITHERSBURG, MARYLAND USA
(301) 258-5011
TELEX: 908736